

DRAGON USER



January 1987

The independent Dragon magazine

Contents

Letters

More machine code looks from Pam D'Arny ... a Cardiff users group... how to double your single sided discs... DOS warning... Chuckle Egg feature

People's Chart

A roundup of the bestselling programs from the past ten months of voting.

News desk

OS-9 license given to H.C. Andersen software... The Thimble's Disk from Ay Software... new prog from Microvics... Quickbeam take over Simultaneous Computing... Christmas offers from Buzz Software.

The Answer

By popular request, Dragon User will from now on be publishing Gordon Lee's own solution to the competition which we are doing in the current issue.

Daisy Wheel

Mike Hales' method for using an ordinary daisywheel printer as a graphics device.

Fractal pictures

All the fractals are here, and the (pre)fractals are telling. At least, they are inside Brian Hulley's Dragon, where self-replicating graphic designs are expanding to fill the screen.

Dragon Soft

8800 Express and Shasta Master from Quickbeam ... Logo 3.4 from Neural Knowledge Software.

Inside The 32

A MacMaster look at the memory and processing features of the Dragon 32, by Dave Barnham B.Sc.

Shorts

Two short programs: Intelligent Patterns Generator and Memory Analyzer

Expert's Arcade Arena

The Expert takes a few pokes at his hapless followers, and drops even more hints about his Christmas present.

Dragon Answers

Brian Cudge takes a look at AT&T functions... REFORMAT in Dragon Basic... gender lead pin code... interrupting printer output... changing disc drives without stopping the program.

Winners and Losers

Gordon Lee looks at some fine examples of the light and shrewd way to win a DU puzzle.

Adventure Trail

Mike Gerrard hits the trail again, this time with the two gentlemen of Verona, and Fernando Pico straight from the author.

Competition

Gordon Lee gets designs on the Dragon with a challenge to come up with the most interesting repeating pattern.

Editorial

A last, quick Happy Christmas to you all if this reaches you when it should, and if it doesn't, then a Happy New Year! I had such a 'good' Christmas last year that I'm thinking of keeping it all to a minimum this year — so you may think of me resucked over my word processor in those snowy, silent days between 28th December and January 1st, putting the next issue into shape. Define other hand — don't phone, I might be at a pump!

There are two or two pieces of interesting news about ... H.C. Andersen at Denmark have been given a license to sell OS-9 in Europe — see News Desk for details. And Quickbeam Software have taken over Simultaneous Computing's software line. The recently silent Microvision has re-emerged with a whole new list of utilities and a few games as well.

In response to many suggestions, Gordon Lee has been rapid in to double his input in Dragon User by extracting some of the better and worse solutions to his puzzles and analysing their success or failure. This month Gordon outlines the principles of effective puzzle-solving. We will also be pointing his solution to the current month's puzzling efforts in future.

Because of a last minute let down, none of the Dragon reporters made it to the London 8800 Show — well the personal reports from anyone who had a good look round. The best ones'll get paid and printed.

How to submit articles

The quality of the material we can publish in Dragon User each month will, to a very great extent depend on the quality of the discoveries that you can make with your Dragon. The Dragon computer was launched prior to the market with a powerful version of Basic, but with very poor documentation.

Articles which are submitted to Dragon User for publication should not be more than 3000 words long. All submissions should be typed. Please leave wide margins and a double space between each line. Programs should, whenever possible, be computer printed on plain white paper and be accompanied by a tape of the program.

We cannot guarantee to return every submit, but article or program, so please keep a copy if you wish. If your program returned you must include a stamped address envelope.

Telephone number
(All departments)
402-0443

Editor
HELEN ARMSTRONG

Production Editor
BARBARA HALE

Associate Editor
JOHN COOK

Editorial Secretary
AMIE MARIE D'AVRYER

Advertisement Manager
RODNEY WOOLLEY

Administration
GILDAIVE SARTTHI

Marketing Manager
HELEN PERRY

Managing Editor
PETER WORLOCK

Publishing Director
JERRY IRELAND

Subscriptions
UK £14 for 13 issues
Overseas (airmail) £30 for 13 issues
ISSN 0264-1171 (Vol. 200275)
Dragon User, 1219 Little Newport Street,
London WC2H 9PP

US address: 20 Business Press International,
200 East 42nd St, New York, NY 10017
Published by BSC Press Ltd
© BSC Press 1987

Typesetting and Production by Artec Limited,
London NW9

Printed by Hadden Brothers Ltd, Ashford, Kent
Registered at the Post Office as a newspaper
Dragon and its logo are trademarks of
Bunhill Ltd

Letters

This is the chance to air your views — send your tips, compliments and complaints to Letters Page, Dragon User, 12-13 Ladbroke New Road, London WC2H 7PP.

Guilt edged

IN OCTOBER 1988 issue you reviewed a program from Casette Games called Casette 90.

I purchased this cassette seven months ago and must entirely agree with your review. In fact I'm surprised you gave it even one dead dragon.

I have to say that I feel guilty about not writing to you at the time to warn people not to buy this complete waste of money, as they say, there's one born every minute!

However, not all is bad, as you reviewed Electronic Author, I must have Printer Control from MacDraw. Consultants which is a nice option, an excellent piece of software which I would have no hesitation in recommending. Your review in a previous issue did not really touch on the excellence and real value for money of this program. Top marks from me.

By the way, apparently Commodore 2's has several severe bugs in it, or so I am informed. If you know about these then the message should have been passed on.

Robt Gardner
21 Newbury Street
Rochester Road
Leicester LE2 5PQ

THE EDITOR has had no first hand complaints about Commodore 2's, but one or two enquiries, and would like to hear from someone who has used the latest version thoroughly ...

Teacher's pet

I HAVE been buying Dragon User since the first issue in 1983, and now have a subscription.

But what do I find? Articles which treat users as morons. Surely there cannot be many (if any) users of the Dragon left who do not know how to program in Basic! The standard of the programs published has not changed in any way since the first issue. I hope they will

be improved, as the machine code articles and programs are of a very good standard. As a computer studies lecturer I see enough poorly thought out pieces of work.

A. D. Butler
Flat 5
18 Wincobale Lane
Meridian
Wolverhampton

NOT AT ALL. Not only are new users still coming to the Dragon, as people pick them up at bargain prices, but we have a steady demand for Basic programs, as well as machine code.

As a teacher with experience of correcting awkward pieces of programming, how about writing an educational piece based on your experience? Drop us a line.

Cardiff club

CALLING all Dragon and Tandy computer users in the Cardiff area, interested in joining a User Group? If so, telephone Mr J. Jones in Cardiff (0303) 553555.

J. Jones
43 Ffrs Ave.
Pontcanna
Cardiff CF1 3PF

Double your disc

HAVING glanced through the July issue I read with interest the article on disc drives by Geoff Elridge. He mentioned that with the Dragon Drive using double sided discs, only one side can be used. This is true using the disc as purchased, but both sides can be used if one uses a paper punch and punches another small hole in the cover on the other side of the centre line to the existing one, keeping the same geometrical distance from the centre. The holes on both sides must be opposite each other for the beam to pass through. Care must be taken so as not to damage the disc. Another must also be cut out on the opposite edge, and in a similar position, to the existing square notch.

I have over a dozen such discs operating in this way with no problems whatsoever. I trust this information can be passed on to the publishing editor for the benefit of other Dragon Drive users.

L. F. Fort
PO Box 190
Llanwrda 410
Paps of South Africa

A week point

I MUST commend you on getting something right. As I'm not referring to your subscription-only policy, I'm referring to your getting my copy of DU here by the second day of the cover month! The June issue actually got here about a week after the May issue. I assume you've taken over from your former distributors. I felt that the problem was there all along and hope this arrangement will continue. I like your magazine very much (almost as much as my Dragon), but was not very happy with the timeline distribution. Your advertisers and writers with whom I've communicated have been exceptionally helpful, too. Thank you.

Donald Hicks
294-51 Emerson St.
Mobile
Alabama AL 36603
USA

THE TRADITION is being preserved, rest assured. Donald. Quite a few of our readers got their September issue only a week after their October one ... the mailing company tell us that they think the labelling database is now sorted correctly, and Anne Marks, who runs the mailing list here, has been awarded an Executive Carlton at Hedges.

Passed his eggsams

I AM desperate to know what the high score for Chuckle-Egg is. The highest score I have ever got is 168893 level 17 after my dad needed the computer.

he replaced the SAM chip. Level 17 has got two ducks and one mud clock.

Jonathan-Raker
Aged 9

Late bookings

MANY thanks to the 'bookspotters' who responded to my article (September issue). Additional titles to add to the convenient space of the bottom of page 14 (Central - Oct. Jan).

The 8089 Companion (Barbara James, Microguide for the Dragon (Virgin/Imbertec) Gifford, 35. Programs for the Dragon32 (Gentry) Langdell, Color Computer Assembly Programming (Radio Shack) Barlen, Getting Started with the T1600/Dragon (Future) Hattwell & Young.

And a correction: Not Programs to Feed your Dragon (Signum by Redwood, Smith and Bacon, not James).

The comment about Color Computer Assembly Programming (Tandy/catalogue number 82-2077) is that it sells, as you would expect, the 80168011 editor/ assembler but is informative and readable. Barlen's 'Colour Computer Graphics' is the one I leap to for instant graphics info, so I can believe that.

Two comments were received on David Barlow's book: 'is generally fine in good, an easier intro than Lerdahl, though less detail' 'It did not get a single program's worth in just about the whole book and I found it too difficult'.

...you pass your money ... Not possessing the book, I couldn't help the reader who asked about the Draughts program in Enter the Dragon. Apparently, it crashes when it reaches the first move, but the text doesn't say so. Can anyone help find through this page, please?

Many thanks to those who took the trouble to write, either to me or to Dragon User, with further book news.

Pam D'Arcy
21 Woodburn Lane
Woodburn Green
High Wycombe
Bucks HP10 5ND

Dragon User People's Chart

The Best Games of 1986

THE final People's Chart ends with a roundup of the games you have voted the most popular in the last ten months. And we have some new plans.

- 1 Juxtaposition.....(Wintersoft)
- 2 Shocktrooper.....(Microdeal)
- 3 Shaolin Master.....(Quickbeam)
- 4 Bean Stalker.....(Micro Vision)
- 5 Speed Racer.....(Microdeal)

THIS IS the Big One — the top five games of 1986, according to your votes. We have a surprise number one — Juxtaposition! Well, you'll be surprised if you never saw Dragon User in your life before — not other

wise. Our mailing, as well as the People's Chart, has borne-bulk witness to the big J's popularity. It's a shame and a pity that Wintersoft had to pack up before they could come up with part II.

Together top four favourites have all pretty well ruled the roost since the People's Chart began in March 1986, hotly pursued by other classics like Jet Set Willy, Boulder Dash, Rastan 3D, Moon Cresta, Total Eclipse (etc.).

This is also the Final Countdown. As features like the Expert's Arcade Arena and Gordon's Last Stand page grow irregularly, we have decided to hand over the reins of the People to the people with their ears to the ground — The Expert and Mike Gerard, on arcade games and adventures respectively. And we'll be making more room for comments from games players on the Letters Page, with star prizes for the best letter every month.

So thank you from us and from the staffware authors to everyone who voted in the People's Chart, keep playing and let us know what you find in the Dragon games world.

The final anagram has to be a good one — well, we think, it's a good one. It goes "Take it from me — (Dragon User is Best)", and its from S. Goode of Sudbury. Your Microdeal software is on its way.

Save to 10% Save Extra Now
Only £4.99 (RST FREE)

THE NEW DE-LUXE TOP VALUE LIGHT PEN FOR THE DRAGON 32/64

THE NEW ADVANCED PROGRAM

for the Trojan light pens include the following facilities —

- DRAW BOX
- DRAW CIRCLE
- DRAW LINE
- DRAW PICTURES FREEHAND
- COLOUR FILL DESIGNATED AREAS
- SAVE AND LOAD PICTURES TO AND FROM TAPE
- FULL ERASE FACILITIES

All in 64-bit colour in any of 4 colours for the Dragon 32/64.

PLUS: FREE voucher entitling you to a 10% DISCOUNT off top Dragon games. (Applies to orders received by 24th December)

Now available from exclusive distributors
PEAKSOFT, 48 Queen Street,
Balderton, Newark, Notts.
NG24 2NS. (Same day mail order service
by phone with credit card)



0438 708200



- DATA ENTRY AND PROCESSING
- MENU SELECTION AND CONTROL
- GAMES PLAYING

This is a full design program which gives hints and tips on how to write programs for the pen, ideal for many educational users.

A top quality pen plus a first class program.
The best value pen package available.

TROJAN

Micro Computer Software & Accessories



News desk

If you have any new products for the Dragon — software or hardware — ring the News Desk on 01-427 4340

OS-9 licence

DANSiI supplies H. G. Andersen Computer A/S have been licensed by Microware in the USA to distribute the OS-9 operating system, Level 1, Version 1.2 for the Dragon.

According to H. G. Andersen, they are the only company to be licensed by Microware to distribute OS-9 for the Dragon.

The Dragon OS-9 system includes edit and assembler, with English manuals, and will cost \$179.00, inc. VAT but excluding delivery charges.

According to Microware, Thomson SIMM of Paris has licensed the OS-980000 operating system for the software nucleus of the recently announced European Educa-

tion Standard microcomputer system.

Three major European electronics companies, Thomson in France, Olivetti in Italy and Agem in the UK are co-operating in developing a new standard in personal and educational micros.

The selection of OS-9 upgrades its increasing acceptance as a standard operating system for 68000 based micros. In late July, Tandy released the Color Computer III based on Microware's system.

Anyone wishing for more information should contact H.G. Andersen Computer A/S at Ringarsvej 380, DK-2770 Rastrup, Denmark.

New, new, new

MICROWARE SOFTWARE, best known to Dragon User readers for reviewer Roy Coates and Brian Staker have announced a new product list of unexpected extensiveness, including games, utilities and support programs, and Pico software.

All the FLEX software comes on disc with documentation, and all the games are available on cassette or DragonDOS disc with the exception of Catacomb Crisis, which is currently only available on cassette says Roy Coates.

The forthcoming list includes the aforementioned Catacomb Crisis (a graphic adventure in caverns and tunnels avoiding spikes and other nasties while looking for treasure, cream cakes (these people have the right idea) and other useful items (C785), Starship Destroyer, Dungeon Destroyer and Wild West Destroyer, three more graphic adventures with location graphics, problems and puzzles (C5000), Commander Compiler, a graphics utility which works with Commander (available from Microware) allowing musical data to be entered on a stand instead of Data statements

(C600), H-Lite, a hi res driver compatible with Dragon Basic, both Basic and machine code driveable (C500), Light Mail, a mailing list which uses a light pen and allows data entry, editing, searching, printing and label production (C500), Pico 1 and 2, which are quite simply collections of Dragon music created on Commander (C300), Help, Plans, Diary, X (an extension of the TSC XOUT utility), and Four (easy access to 4Space printers) all on one disc (C700), Icons, which re-routes all IO through the PICO2 port to allow a terminal to be connected up (C300), Game of Chance (already reviewed), 4Ks of FLEX classic adventure (C700), Shellcode, a FLEX utility which allows sector editing, mapping, tracing and dumping (C600) and CMDR, a FLEX programmers' utility which allows ASCII or hex dumps, memory searches etc (C700) and (available mid December) EDIT, a re configuration FLEX screen editor (price to be announced, but it won't be cheap).

More information from Microware Software, 8 Wensley Road, Liverpool L8 9DW



Shaolin Master from Quickbeam

Quickbeam quicker

DAVE HITCHMAN has been in touch to tell us that as from 23rd November, Smithson Computing will become part of Quickbeam Software, ceasing to operate as a separate enterprise.

Wayne Smithson is hoping to do some programming for Quickbeam.

Dragon Monthly subscribers will be glad to hear that the newsletter will continue with Quickbeam under the new name of Dragon's Tail, run by

Julian Cogdell, 34 Shooter's Drive, Mawley, Essex EN9 9DD. All previous subscriptions will be honoured, says Dave.

Smithson Computing's popular single which included Gordon Bennett and Electronic Author, as well as a new game, 100 Screen Frenzy, will join Quickbeam's list.

For more details contact David Hitchman, 67 Old Naxosing Road, Broxbourne, Herts EN10 8RN.

Lucky thirteenth

ALAN Cook of Arc Software has been on lucky's list that The Thirteenth Ball (reviewed by Mike Gerrard in December's Adventure Tail) is now out and ready for sale and has been picking up favourable reviews in the shops. Mike liked it. The game costs £3 from Arc Software, 273 Moira Road, Newton Mearns, Glasgow G17 5LY.

Buzz offer

Buzz Software have a number of pre-Christmas special offers, including Small Business Telephone Directory and Small

Business Mail Processor. For further information, send them a T10 stamp for their information pack.

Write to: Buzz Software, 15 St Stephen Road, Penketh, Nr. Warrington, Cheshire WA9 2RN.

Quickbeam still quick

QU wishes to apologise to anyone who has waited more than 26 days for their copies of FireForce and would like to see that any hold up has been getting the order through the Dragon User filing system, and not through any fault of Quickbeam's. All orders received before the closing date have now been dealt with.

The Answer

This is Gordon Lee's own
solution to the October competition
see page 50 for results

We have had a number of requests for Gordon Lee's solution to his monthly competitions to be published. As there is no 'right' way to find the answer to most of his puzzles, we felt that it would be rather misleading to suggest that these were the 'proper' solutions, at the expense of someone else's equally effective program.

Many people have said that it would be a useful programming guide for them to see Gordon Lee's own answer so, we have decided to take a close look at answers all round, and accordingly are publishing here Gordon's solution to the OCTOBER competition.

Professor Hirst's solution was:

1 6 5 3 2
3 1 8 3 6
6 3 2 5 1
8 3 5 1 2
5 6 3 2 1

2 2 1 1 6 2

Involved E=1, L=6, G=5, A=3 and R=2.

Solution: the program as listed computes all possible substitutions of digits for the five letters used in the sum. (Note that, with alphacrypt of this type, leading zeroes are not allowed, so E, L, G and R cannot be zero.)

Once this is done, it constructs the five relevant values of the words in the sum. As

each digit is found its letter equivalent is added to string WS to reconstruct the letter sequence in the answer.

These values are also stored in array WSQ for reference. If all possible letter sequences are printed out (see enclosed list), the results can be compared with the two re-arrangements. As it was stated that neither of

these has any letter correctly placed, it is possible to cross off words with letters that match any letters in those key words. For example, all sequences with either an L or an G in first place can be crossed off.

This leaves just one value FIVEEIGH, and its generating value (66332) can be obtained from array WSQ.

```

10 DIM WS(20)
20 DIM WSQ(20)
30 DIM WSQ2(20)
40 DIM WSQ3(20)
50 DIM WSQ4(20)
60 DIM WSQ5(20)
70 DIM WSQ6(20)
80 DIM WSQ7(20)
90 DIM WSQ8(20)
100 DIM WSQ9(20)
110 DIM WSQ10(20)
120 DIM WSQ11(20)
130 DIM WSQ12(20)
140 DIM WSQ13(20)
150 DIM WSQ14(20)
160 DIM WSQ15(20)
170 DIM WSQ16(20)
180 DIM WSQ17(20)
190 DIM WSQ18(20)
200 DIM WSQ19(20)
210 DIM WSQ20(20)
220 DIM WSQ21(20)
230 DIM WSQ22(20)
240 DIM WSQ23(20)
250 DIM WSQ24(20)
260 DIM WSQ25(20)
270 DIM WSQ26(20)
280 DIM WSQ27(20)
290 DIM WSQ28(20)
300 DIM WSQ29(20)
310 DIM WSQ30(20)
320 DIM WSQ31(20)
330 DIM WSQ32(20)
340 DIM WSQ33(20)
350 DIM WSQ34(20)
360 DIM WSQ35(20)
370 DIM WSQ36(20)
380 DIM WSQ37(20)
390 DIM WSQ38(20)
400 DIM WSQ39(20)
410 DIM WSQ40(20)
420 DIM WSQ41(20)
430 DIM WSQ42(20)
440 DIM WSQ43(20)
450 DIM WSQ44(20)
460 DIM WSQ45(20)
470 DIM WSQ46(20)
480 DIM WSQ47(20)
490 DIM WSQ48(20)
500 DIM WSQ49(20)
510 DIM WSQ50(20)
520 DIM WSQ51(20)
530 DIM WSQ52(20)
540 DIM WSQ53(20)
550 DIM WSQ54(20)
560 DIM WSQ55(20)
570 DIM WSQ56(20)
580 DIM WSQ57(20)
590 DIM WSQ58(20)
600 DIM WSQ59(20)
610 DIM WSQ60(20)
620 DIM WSQ61(20)
630 DIM WSQ62(20)
640 DIM WSQ63(20)
650 DIM WSQ64(20)
660 DIM WSQ65(20)
670 DIM WSQ66(20)
680 DIM WSQ67(20)
690 DIM WSQ68(20)
700 DIM WSQ69(20)
710 DIM WSQ70(20)
720 DIM WSQ71(20)
730 DIM WSQ72(20)
740 DIM WSQ73(20)
750 DIM WSQ74(20)
760 DIM WSQ75(20)
770 DIM WSQ76(20)
780 DIM WSQ77(20)
790 DIM WSQ78(20)
800 DIM WSQ79(20)
810 DIM WSQ80(20)
820 DIM WSQ81(20)
830 DIM WSQ82(20)
840 DIM WSQ83(20)
850 DIM WSQ84(20)
860 DIM WSQ85(20)
870 DIM WSQ86(20)
880 DIM WSQ87(20)
890 DIM WSQ88(20)
900 DIM WSQ89(20)
910 DIM WSQ90(20)
920 DIM WSQ91(20)
930 DIM WSQ92(20)
940 DIM WSQ93(20)
950 DIM WSQ94(20)
960 DIM WSQ95(20)
970 DIM WSQ96(20)
980 DIM WSQ97(20)
990 DIM WSQ98(20)
1000 DIM WSQ99(20)
1010 DIM WSQ100(20)
1020 DIM WSQ101(20)
1030 DIM WSQ102(20)
1040 DIM WSQ103(20)
1050 DIM WSQ104(20)
1060 DIM WSQ105(20)
1070 DIM WSQ106(20)
1080 DIM WSQ107(20)
1090 DIM WSQ108(20)
1100 DIM WSQ109(20)
1110 DIM WSQ110(20)
1120 DIM WSQ111(20)
1130 DIM WSQ112(20)
1140 DIM WSQ113(20)
1150 DIM WSQ114(20)
1160 DIM WSQ115(20)
1170 DIM WSQ116(20)
1180 DIM WSQ117(20)
1190 DIM WSQ118(20)
1200 DIM WSQ119(20)
1210 DIM WSQ120(20)
1220 DIM WSQ121(20)
1230 DIM WSQ122(20)
1240 DIM WSQ123(20)
1250 DIM WSQ124(20)
1260 DIM WSQ125(20)
1270 DIM WSQ126(20)
1280 DIM WSQ127(20)
1290 DIM WSQ128(20)
1300 DIM WSQ129(20)
1310 DIM WSQ130(20)
1320 DIM WSQ131(20)
1330 DIM WSQ132(20)
1340 DIM WSQ133(20)
1350 DIM WSQ134(20)
1360 DIM WSQ135(20)
1370 DIM WSQ136(20)
1380 DIM WSQ137(20)
1390 DIM WSQ138(20)
1400 DIM WSQ139(20)
1410 DIM WSQ140(20)
1420 DIM WSQ141(20)
1430 DIM WSQ142(20)
1440 DIM WSQ143(20)
1450 DIM WSQ144(20)
1460 DIM WSQ145(20)
1470 DIM WSQ146(20)
1480 DIM WSQ147(20)
1490 DIM WSQ148(20)
1500 DIM WSQ149(20)
1510 DIM WSQ150(20)
1520 DIM WSQ151(20)
1530 DIM WSQ152(20)
1540 DIM WSQ153(20)
1550 DIM WSQ154(20)
1560 DIM WSQ155(20)
1570 DIM WSQ156(20)
1580 DIM WSQ157(20)
1590 DIM WSQ158(20)
1600 DIM WSQ159(20)
1610 DIM WSQ160(20)
1620 DIM WSQ161(20)
1630 DIM WSQ162(20)
1640 DIM WSQ163(20)
1650 DIM WSQ164(20)
1660 DIM WSQ165(20)
1670 DIM WSQ166(20)
1680 DIM WSQ167(20)
1690 DIM WSQ168(20)
1700 DIM WSQ169(20)
1710 DIM WSQ170(20)
1720 DIM WSQ171(20)
1730 DIM WSQ172(20)
1740 DIM WSQ173(20)
1750 DIM WSQ174(20)
1760 DIM WSQ175(20)
1770 DIM WSQ176(20)
1780 DIM WSQ177(20)
1790 DIM WSQ178(20)
1800 DIM WSQ179(20)
1810 DIM WSQ180(20)
1820 DIM WSQ181(20)
1830 DIM WSQ182(20)
1840 DIM WSQ183(20)
1850 DIM WSQ184(20)
1860 DIM WSQ185(20)
1870 DIM WSQ186(20)
1880 DIM WSQ187(20)
1890 DIM WSQ188(20)
1900 DIM WSQ189(20)
1910 DIM WSQ190(20)
1920 DIM WSQ191(20)
1930 DIM WSQ192(20)
1940 DIM WSQ193(20)
1950 DIM WSQ194(20)
1960 DIM WSQ195(20)
1970 DIM WSQ196(20)
1980 DIM WSQ197(20)
1990 DIM WSQ198(20)
2000 DIM WSQ199(20)
2010 DIM WSQ200(20)
2020 DIM WSQ201(20)
2030 DIM WSQ202(20)
2040 DIM WSQ203(20)
2050 DIM WSQ204(20)
2060 DIM WSQ205(20)
2070 DIM WSQ206(20)
2080 DIM WSQ207(20)
2090 DIM WSQ208(20)
2100 DIM WSQ209(20)
2110 DIM WSQ210(20)
2120 DIM WSQ211(20)
2130 DIM WSQ212(20)
2140 DIM WSQ213(20)
2150 DIM WSQ214(20)
2160 DIM WSQ215(20)
2170 DIM WSQ216(20)
2180 DIM WSQ217(20)
2190 DIM WSQ218(20)
2200 DIM WSQ219(20)
2210 DIM WSQ220(20)
2220 DIM WSQ221(20)
2230 DIM WSQ222(20)
2240 DIM WSQ223(20)
2250 DIM WSQ224(20)
2260 DIM WSQ225(20)
2270 DIM WSQ226(20)
2280 DIM WSQ227(20)
2290 DIM WSQ228(20)
2300 DIM WSQ229(20)
2310 DIM WSQ230(20)
2320 DIM WSQ231(20)
2330 DIM WSQ232(20)
2340 DIM WSQ233(20)
2350 DIM WSQ234(20)
2360 DIM WSQ235(20)
2370 DIM WSQ236(20)
2380 DIM WSQ237(20)
2390 DIM WSQ238(20)
2400 DIM WSQ239(20)
2410 DIM WSQ240(20)
2420 DIM WSQ241(20)
2430 DIM WSQ242(20)
2440 DIM WSQ243(20)
2450 DIM WSQ244(20)
2460 DIM WSQ245(20)
2470 DIM WSQ246(20)
2480 DIM WSQ247(20)
2490 DIM WSQ248(20)
2500 DIM WSQ249(20)
2510 DIM WSQ250(20)
2520 DIM WSQ251(20)
2530 DIM WSQ252(20)
2540 DIM WSQ253(20)
2550 DIM WSQ254(20)
2560 DIM WSQ255(20)
2570 DIM WSQ256(20)
2580 DIM WSQ257(20)
2590 DIM WSQ258(20)
2600 DIM WSQ259(20)
2610 DIM WSQ260(20)
2620 DIM WSQ261(20)
2630 DIM WSQ262(20)
2640 DIM WSQ263(20)
2650 DIM WSQ264(20)
2660 DIM WSQ265(20)
2670 DIM WSQ266(20)
2680 DIM WSQ267(20)
2690 DIM WSQ268(20)
2700 DIM WSQ269(20)
2710 DIM WSQ270(20)
2720 DIM WSQ271(20)
2730 DIM WSQ272(20)
2740 DIM WSQ273(20)
2750 DIM WSQ274(20)
2760 DIM WSQ275(20)
2770 DIM WSQ276(20)
2780 DIM WSQ277(20)
2790 DIM WSQ278(20)
2800 DIM WSQ279(20)
2810 DIM WSQ280(20)
2820 DIM WSQ281(20)
2830 DIM WSQ282(20)
2840 DIM WSQ283(20)
2850 DIM WSQ284(20)
2860 DIM WSQ285(20)
2870 DIM WSQ286(20)
2880 DIM WSQ287(20)
2890 DIM WSQ288(20)
2900 DIM WSQ289(20)
2910 DIM WSQ290(20)
2920 DIM WSQ291(20)
2930 DIM WSQ292(20)
2940 DIM WSQ293(20)
2950 DIM WSQ294(20)
2960 DIM WSQ295(20)
2970 DIM WSQ296(20)
2980 DIM WSQ297(20)
2990 DIM WSQ298(20)
3000 DIM WSQ299(20)
3010 DIM WSQ300(20)
3020 DIM WSQ301(20)
3030 DIM WSQ302(20)
3040 DIM WSQ303(20)
3050 DIM WSQ304(20)
3060 DIM WSQ305(20)
3070 DIM WSQ306(20)
3080 DIM WSQ307(20)
3090 DIM WSQ308(20)
3100 DIM WSQ309(20)
3110 DIM WSQ310(20)
3120 DIM WSQ311(20)
3130 DIM WSQ312(20)
3140 DIM WSQ313(20)
3150 DIM WSQ314(20)
3160 DIM WSQ315(20)
3170 DIM WSQ316(20)
3180 DIM WSQ317(20)
3190 DIM WSQ318(20)
3200 DIM WSQ319(20)
3210 DIM WSQ320(20)
3220 DIM WSQ321(20)
3230 DIM WSQ322(20)
3240 DIM WSQ323(20)
3250 DIM WSQ324(20)
3260 DIM WSQ325(20)
3270 DIM WSQ326(20)
3280 DIM WSQ327(20)
3290 DIM WSQ328(20)
3300 DIM WSQ329(20)
3310 DIM WSQ330(20)
3320 DIM WSQ331(20)
3330 DIM WSQ332(20)
3340 DIM WSQ333(20)
3350 DIM WSQ334(20)
3360 DIM WSQ335(20)
3370 DIM WSQ336(20)
3380 DIM WSQ337(20)
3390 DIM WSQ338(20)
3400 DIM WSQ339(20)
3410 DIM WSQ340(20)
3420 DIM WSQ341(20)
3430 DIM WSQ342(20)
3440 DIM WSQ343(20)
3450 DIM WSQ344(20)
3460 DIM WSQ345(20)
3470 DIM WSQ346(20)
3480 DIM WSQ347(20)
3490 DIM WSQ348(20)
3500 DIM WSQ349(20)
3510 DIM WSQ350(20)
3520 DIM WSQ351(20)
3530 DIM WSQ352(20)
3540 DIM WSQ353(20)
3550 DIM WSQ354(20)
3560 DIM WSQ355(20)
3570 DIM WSQ356(20)
3580 DIM WSQ357(20)
3590 DIM WSQ358(20)
3600 DIM WSQ359(20)
3610 DIM WSQ360(20)
3620 DIM WSQ361(20)
3630 DIM WSQ362(20)
3640 DIM WSQ363(20)
3650 DIM WSQ364(20)
3660 DIM WSQ365(20)
3670 DIM WSQ366(20)
3680 DIM WSQ367(20)
3690 DIM WSQ368(20)
3700 DIM WSQ369(20)
3710 DIM WSQ370(20)
3720 DIM WSQ371(20)
3730 DIM WSQ372(20)
3740 DIM WSQ373(20)
3750 DIM WSQ374(20)
3760 DIM WSQ375(20)
3770 DIM WSQ376(20)
3780 DIM WSQ377(20)
3790 DIM WSQ378(20)
3800 DIM WSQ379(20)
3810 DIM WSQ380(20)
3820 DIM WSQ381(20)
3830 DIM WSQ382(20)
3840 DIM WSQ383(20)
3850 DIM WSQ384(20)
3860 DIM WSQ385(20)
3870 DIM WSQ386(20)
3880 DIM WSQ387(20)
3890 DIM WSQ388(20)
3900 DIM WSQ389(20)
3910 DIM WSQ390(20)
3920 DIM WSQ391(20)
3930 DIM WSQ392(20)
3940 DIM WSQ393(20)
3950 DIM WSQ394(20)
3960 DIM WSQ395(20)
3970 DIM WSQ396(20)
3980 DIM WSQ397(20)
3990 DIM WSQ398(20)
4000 DIM WSQ399(20)
4010 DIM WSQ400(20)
4020 DIM WSQ401(20)
4030 DIM WSQ402(20)
4040 DIM WSQ403(20)
4050 DIM WSQ404(20)
4060 DIM WSQ405(20)
4070 DIM WSQ406(20)
4080 DIM WSQ407(20)
4090 DIM WSQ408(20)
4100 DIM WSQ409(20)
4110 DIM WSQ410(20)
4120 DIM WSQ411(20)
4130 DIM WSQ412(20)
4140 DIM WSQ413(20)
4150 DIM WSQ414(20)
4160 DIM WSQ415(20)
4170 DIM WSQ416(20)
4180 DIM WSQ417(20)
4190 DIM WSQ418(20)
4200 DIM WSQ419(20)
4210 DIM WSQ420(20)
4220 DIM WSQ421(20)
4230 DIM WSQ422(20)
4240 DIM WSQ423(20)
4250 DIM WSQ424(20)
4260 DIM WSQ425(20)
4270 DIM WSQ426(20)
4280 DIM WSQ427(20)
4290 DIM WSQ428(20)
4300 DIM WSQ429(20)
4310 DIM WSQ430(20)
4320 DIM WSQ431(20)
4330 DIM WSQ432(20)
4340 DIM WSQ433(20)
4350 DIM WSQ434(20)
4360 DIM WSQ435(20)
4370 DIM WSQ436(20)
4380 DIM WSQ437(20)
4390 DIM WSQ438(20)
4400 DIM WSQ439(20)
4410 DIM WSQ440(20)
4420 DIM WSQ441(20)
4430 DIM WSQ442(20)
4440 DIM WSQ443(20)
4450 DIM WSQ444(20)
4460 DIM WSQ445(20)
4470 DIM WSQ446(20)
4480 DIM WSQ447(20)
4490 DIM WSQ448(20)
4500 DIM WSQ449(20)
4510 DIM WSQ450(20)
4520 DIM WSQ451(20)
4530 DIM WSQ452(20)
4540 DIM WSQ453(20)
4550 DIM WSQ454(20)
4560 DIM WSQ455(20)
4570 DIM WSQ456(20)
4580 DIM WSQ457(20)
4590 DIM WSQ458(20)
4600 DIM WSQ459(20)
4610 DIM WSQ460(20)
4620 DIM WSQ461(20)
4630 DIM WSQ462(20)
4640 DIM WSQ463(20)
4650 DIM WSQ464(20)
4660 DIM WSQ465(20)
4670 DIM WSQ466(20)
4680 DIM WSQ467(20)
4690 DIM WSQ468(20)
4700 DIM WSQ469(20)
4710 DIM WSQ470(20)
4720 DIM WSQ471(20)
4730 DIM WSQ472(20)
4740 DIM WSQ473(20)
4750 DIM WSQ474(20)
4760 DIM WSQ475(20)
4770 DIM WSQ476(20)
4780 DIM WSQ477(20)
4790 DIM WSQ478(20)
4800 DIM WSQ479(20)
4810 DIM WSQ480(20)
4820 DIM WSQ481(20)
4830 DIM WSQ482(20)
4840 DIM WSQ483(20)
4850 DIM WSQ484(20)
4860 DIM WSQ485(20)
4870 DIM WSQ486(20)
4880 DIM WSQ487(20)
4890 DIM WSQ488(20)
4900 DIM WSQ489(20)
4910 DIM WSQ490(20)
4920 DIM WSQ491(20)
4930 DIM WSQ492(20)
4940 DIM WSQ493(20)
4950 DIM WSQ494(20)
4960 DIM WSQ495(20)
4970 DIM WSQ496(20)
4980 DIM WSQ497(20)
4990 DIM WSQ498(20)
5000 DIM WSQ499(20)
5010 DIM WSQ500(20)
5020 DIM WSQ501(20)
5030 DIM WSQ502(20)
5040 DIM WSQ503(20)
5050 DIM WSQ504(20)
5060 DIM WSQ505(20)
5070 DIM WSQ506(20)
5080 DIM WSQ507(20)
5090 DIM WSQ508(20)
5100 DIM WSQ509(20)
5110 DIM WSQ510(20)
5120 DIM WSQ511(20)
5130 DIM WSQ512(20)
5140 DIM WSQ513(20)
5150 DIM WSQ514(20)
5160 DIM WSQ515(20)
5170 DIM WSQ516(20)
5180 DIM WSQ517(20)
5190 DIM WSQ518(20)
5200 DIM WSQ519(20)
5210 DIM WSQ520(20)
5220 DIM WSQ521(20)
5230 DIM WSQ522(20)
5240 DIM WSQ523(20)
5250 DIM WSQ524(20)
5260 DIM WSQ525(20)
5270 DIM WSQ526(20)
5280 DIM WSQ527(20)
5290 DIM WSQ528(20)
5300 DIM WSQ529(20)
5310 DIM WSQ530(20)
5320 DIM WSQ531(20)
5330 DIM WSQ532(20)
5340 DIM WSQ533(20)
5350 DIM WSQ534(20)
5360 DIM WSQ535(20)
5370 DIM WSQ536(20)
5380 DIM WSQ537(20)
5390 DIM WSQ538(20)
5400 DIM WSQ539(20)
5410 DIM WSQ540(20)
5420 DIM WSQ541(20)
5430 DIM WSQ542(20)
5440 DIM WSQ543(20)
5450 DIM WSQ544(20)
5460 DIM WSQ545(20)
5470 DIM WSQ546(20)
5480 DIM WSQ547(20)
5490 DIM WSQ548(20)
5500 DIM WSQ549(20)
5510 DIM WSQ550(20)
5520 DIM WSQ551(20)
5530 DIM WSQ552(20)
5540 DIM WSQ553(20)
5550 DIM WSQ554(20)
5560 DIM WSQ555(20)
5570 DIM WSQ556(20)
5580 DIM WSQ557(20)
5590 DIM WSQ558(20)
5600 DIM WSQ559(20)
5610 DIM WSQ560(20)
5620 DIM WSQ561(20)
5630 DIM WSQ562(20)
5640 DIM WSQ563(20)
5650 DIM WSQ564(20)
5660 DIM WSQ565(20)
5670 DIM WSQ566(20)
5680 DIM WSQ567(20)
5690 DIM WSQ568(20)
5700 DIM WSQ569(20)
5710 DIM WSQ570(20)
5720 DIM WSQ571(20)
5730 DIM WSQ572(20)
5740 DIM WSQ573(20)
5750 DIM WSQ574(20)
5760 DIM WSQ575(20)
5770 DIM WSQ576(20)
5780 DIM WSQ577(20)
5790 DIM WSQ578(20)
5800 DIM WSQ579(20)
5810 DIM WSQ580(20)
5820 DIM WSQ581(20)
5830 DIM WSQ582(20)
5840 DIM WSQ583(20)
5850 DIM WSQ584(20)
5860 DIM WSQ585(20)
5870 DIM WSQ586(20)
5880 DIM WSQ587(20)
5890 DIM WSQ588(20)
5900 DIM WSQ589(20)
5910 DIM WSQ590(20)
5920 DIM WSQ591(20)
5930 DIM WSQ592(20)
5940 DIM WSQ593(20)
5950 DIM WSQ594(20)
5960 DIM WSQ595(20)
5970 DIM WSQ596(20)
5980 DIM WSQ597(20)
5990 DIM WSQ598(20)
6000 DIM WSQ599(20)
6010 DIM WSQ600(20)
6020 DIM WSQ601(20)
6030 DIM WSQ602(20)
6040 DIM WSQ603(20)
6050 DIM WSQ604(20)
6060 DIM WSQ605(20)
6070 DIM WSQ606(20)
6080 DIM WSQ607(20)
6090 DIM WSQ608(20)
6100 DIM WSQ609(20)
6110 DIM WSQ610(20)
6120 DIM WSQ611(20)
6130 DIM WSQ612(20)
6140 DIM WSQ613(20)
6150 DIM WSQ614(20)
6160 DIM WSQ615(20)
6170 DIM WSQ616(20)
6180 DIM WSQ617(20)
6190 DIM WSQ618(20)
6200 DIM WSQ619(20)
6210 DIM WSQ620(20)
6220 DIM WSQ621(20)
6230 DIM WSQ622(20)
6240 DIM WSQ623(20)
6250 DIM WSQ624(20)
6260 DIM WSQ625(20)
6270 DIM WSQ626(20)
6280 DIM WSQ627(20)
6290 DIM WSQ628(20)
6300 DIM WSQ629(20)
6310 DIM WSQ630(20)
6320 DIM WSQ631(20)
6330 DIM WSQ632(20)
6340 DIM WSQ633(20)
6350 DIM WSQ634(20)
6360 DIM WSQ635(20)
6370 DIM WSQ636(20)
6380 DIM WSQ637(20)
6390 DIM WSQ638(20)
6400 DIM WSQ639(20)
6410 DIM WSQ640(20)
6420 DIM WSQ641(20)
6430 DIM WSQ642(20)
6440 DIM WSQ643(20)
6450 DIM WSQ644(20)
6460 DIM WSQ645(20)
6470 DIM WSQ646(20)
6480 DIM WSQ647(20)
6490 DIM WSQ648(20)
6500 DIM WSQ649(20)
6510 DIM WSQ650(20)
6520 DIM WSQ651(20)
6530 DIM WSQ652(20)
6540 DIM WSQ653(20)
6550 DIM WSQ654(20)
6560 DIM WSQ655(20)
6570 DIM WSQ656(20)
6580 DIM WSQ657(20)
6590 DIM WSQ658(20)
6600 DIM WSQ659(20)
6610 DIM WSQ660(20)
6620 DIM WSQ661(20)
6630 DIM WSQ662(20)
6640 DIM WSQ663(20)
6650 DIM WSQ664(20)
6660 DIM WSQ665(20)
6670 DIM WSQ666(20)
6680 DIM WSQ667(20)
6690 DIM WSQ668(20)
6700 DIM WSQ669(20)
6710 DIM WSQ670(20)
6720 DIM WSQ671(20)
6730 DIM WSQ672(20)
6740 DIM WSQ673(20)
6750 DIM WSQ674(20)
6760 DIM WSQ675(20)
6770 DIM WSQ676(20)
6780 DIM WSQ677(20)
6790 DIM WSQ678(20)
6800 DIM WSQ679(20)
6810 DIM WSQ680(20)
6820 DIM WSQ681(20)
6830 DIM WSQ682(20)
6840 DIM WSQ683(20)
6850 DIM WSQ684(20)
6860 DIM WSQ685(20)
6870 DIM WSQ686(20)
6880 DIM WSQ687(20)
6890 DIM WSQ688(20)
6900 DIM WSQ689(20)
6910 DIM WSQ690(20)
6920 DIM WSQ691(20)
6930 DIM WSQ692(20)
6940 DIM WSQ693(20)
6950 DIM WSQ694(20)
6960 DIM WSQ695(20)
6970 DIM WSQ696(20)
6980 DIM WSQ697(20)
6990 DIM WSQ698(20)
7000 DIM WSQ699(20)
7010 DIM WSQ700(20)
7020 DIM WSQ701(20)
7030 DIM WSQ702(20)
7040 DIM WSQ703(20)
7050 DIM WSQ704(20)
7060 DIM WSQ705(20)
7070 DIM WSQ706(20)
7080 DIM WSQ707(20)
7090 DIM WSQ708(20)
7100 DIM WSQ709(20)
7110 DIM WSQ710(20)
7120 DIM WSQ711(20)
7130 DIM WSQ712(20)
7140 DIM WSQ713(20)
7150 DIM WSQ714(20)
7160 DIM WSQ715(20)
7170 DIM WSQ716(20)
7180 DIM WSQ717(20)
7190 DIM WSQ718(20)
7200 DIM WSQ719(20)
7210 DIM WSQ720(20)
7220 DIM WSQ721(20)
7230 DIM WSQ722(20)
7240 DIM WSQ723(20)
7250 DIM WSQ724(20)
7260 DIM WSQ725(20)
7270 DIM WSQ726(20)
7280 DIM WSQ727(20)
7290 DIM WSQ728(20)
7300 DIM WSQ729(20)
7310 DIM WSQ730(20)
7320 DIM WSQ731(20)
7330 DIM WSQ732(20)
7340 DIM WSQ733(20)
7350 DIM WSQ734(20)
7360 DIM WSQ735(20)
7370 DIM WSQ736(20)
7380 DIM WSQ737(20)
7390 DIM WSQ738(20)
7400 DIM WSQ739(20)
7410 DIM WSQ740(20)
7420 DIM WSQ741(20)
7430 DIM WSQ742(20)
7440 DIM WSQ743(20)
7450 DIM WSQ744(20)
7460 DIM WSQ745(20)
7470 DIM WSQ746(20)
7480 DIM WSQ747(20)
7490 DIM WSQ748(20)
7500 DIM WSQ749(20)
7510 DIM WSQ750(20)
7520 DIM WSQ751(20)
7530 DIM WSQ752(20)
7540 DIM WSQ753(20)
7550 DIM WSQ754(20)
7560 DIM WSQ755(20)
7570 DIM WSQ756(20)
7580 DIM WSQ757(20)
7590 DIM WSQ758(20)
7600 DIM WSQ759(20)
7610 DIM WSQ760(20)
7620 DIM WSQ761(20)
7630 DIM WSQ762(20)
7640 DIM WSQ763(20)
7650 DIM WSQ764(20)
7660 DIM WSQ765(20)
7670 DIM WSQ766(20)
7680 DIM WSQ767(20)
7690 DIM WSQ768(20)
7700 DIM WSQ769(20)
7710 DIM WSQ770(20)
7720 DIM WSQ771(20)
7730 DIM WSQ772(20)
7740 DIM WSQ773(20)
7750 DIM WSQ774(20)
7760 DIM WSQ7
```

Daisywheel graphics

Learn how to get squares and circles out of your printer, says Mike Hosken

THERE seems to be a supposition in some quarters that only plotters and dot-matrix printers can produce printed graphics. But a daisywheel is equipped with a perfectly good dot — the full stop.

Depended the character width and line depth can be suitably adjusted there is no reason why a program cannot be devised to dump screen graphics onto paper, for example.

Squares, squares and round circles — needs five 120ths (ie, one 24th of an inch) by two 48ths (which is also one 24th of an inch). But according to the printer instruction books the codes need, for some unspecified reason, to be one greater than the number actually required, so it's code 49 for character width, and code 5 for line depth.

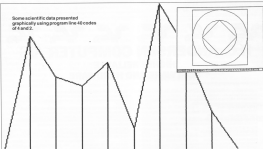
A screen dump can be incorporated into any graphics program. But it doesn't have to.

Like Left/Right, it could not be necessary to emphasize the vital importance of the punctuation in line 80. But line 100 is not essential: it simply re-initialises the printer, setting character and line dimensions back to normal as if you had switched the printer off and back on again. CHR\$(13) just prints the paper right out for your added convenience.

255 characters each one 24th of an inch

```
10 REM High resolution screen dump for daisywheel printer.
20 CLS: PRINT "CHECK THAT THE PRINTER IS READY AND THEN PRESS 'ENTER'..."
30 INPUT A$: SCREEN 1,1
40 PRINT #=2,CHR$(27)CHR$(31)CHR$(6)CHR$(27)CHR$(30)CHR$(5)
50 FOR UD=0 TO 191
60 FOR LR=0 TO 255
70 IF PPOINT(LR,UD)=0 THEN PRINT #=2," "; ELSE PRINT #=2,".";
80 NEXT LR
90 PRINT #=2: NEXT UD
100 PRINT #=2,CHR$(27)CHR$(26)CHR$(13)CHR$(12)
```

Some scientific data presented graphically using program line 48 codes of 4 and 5.



Among the commoner daisywheels at the lower end of the market seem to be those which are marketed as Daisystep 2000, MicroP and Quendata. Sending CHR\$(6) codes can set the character width to anything from zero to any number of one-hundred and twentieths of an inch, or line depth similarly in units of forty-sixths of an inch. So to get geometric accuracy —

let's say you have a program and produced the left-hand plot in art or design the masterpiece will be retained in graphics memory even after MEM and loading a separate graphics dump program. But don't switch off in between, not use PCLS, nor re-assign memory with a POKEVAR.

In pretty well its simplest form, a suitable SFD here, UD is the Up/Down coordinate,

wide amount is something under eleven inches, firing nicely onto A4 paper sideways. But 180 lines each one 24th of an inch deep come to just eight inches, requiring that the paper be very carefully inserted to give only a tenth of an inch top margin. Smaller versions can be produced by altering the 6 and 3 codes in line 40 — if the resulting geometrical distortion doesn't matter.

CRAZY SALE

EXCLUSIVE
TO DRAGON USER

OF INCENTIVE BLOCKBUSTERS

ARCADE

BACKTRACK
DRAGON USER STAR GAME
— 5 DRAGON RATING —
Normal RRP £6.50
CRAZY SALE
PRICE
£2.95

STAR
GAME

ADVENTURE

THE KETTRILOGY
3 ADVENTURE FOR THE
PRICE OF ONE!
Normal RRP £3.95
CRAZY SALE PRICE
£3.95

EDDIE

STEADY GO

DRAGON USER
5 DRAGON RATING
ARCADE 66 LEVELS
Normal RRP £6.95
CRAZY PRICE SALE
£3.95

MOON

CRESTA

5 DRAGON RATING AGAIN!
THE OFFICIALLY LICENCED
SHOOT EM UP CLASSIC
Normal RRP £7.95
CRAZY SALE PRICE
£3.95

ARCADE

ARCADE

CRAZY SALE ORDER FORM



Please rush me (1st Class Post)

Backtrack £2.95 ☐

Kettrilogy £3.95 ☐

Moon Cresta £3.95 ☐

Eddie Steady Go £3.95 ☐

I enclose my cheque _____ please debit card no. _____

Name / Address _____

Incentive Software, 2 Minerva House, Calleva Park, Aldermaston,
Berkshire RG7 4QW.

Orders only accepted with this form or photocopy received by 31st
January 1987.

0100

Fractal pictures

Brian Hulley describes the notion of fractals and tree growth

UP until recently I was under the impression that you would have to use a mainframe to generate fractal pictures. However I soon realised that it is perfectly possible to generate fractals using the Dragon. For anyone who has escaped the flood of interest in fractals, they are simply structures which are formed when one basic shape (the initiator) can be used to derive several secondary shapes by means of a generator. For example, in figure one, a vertical line splits into two branches which then split again and again to give the image of a tree in figure five. By using the computer, we can simulate this repeated division and take the process much further than four divisions to derive a more realistic picture. In doing this the computer is not only creating a picture, it is also simulating the growth of a real tree. By then introducing a random element to the program, an even more accurate simulation of natural growth can be achieved.

The first program uses the process of repeated division to draw a tree. The formula derived in figure five is used to obtain the end coordinates of two secondary branches given the end coordinates of the primary branch and the angle this makes with the horizontal (to eliminate 90 errors). The length of the primary branch by the branching ratio, and the angle that each secondary branch makes with the primary, is the branching angle, θ . (The other branch is derived by substituting $-\theta$ into the equation).

Randomness is introduced in the second program, by adding two new variables, R and Z. Z contains the constant branching number and R contains the random part.

```
U=(Z-RND)/100
% to modify the first program to get the second version, simply insert:
145 U=(Z-RND)/100
445 INPUT "BRANCHING OFFSET (R) " R
Then change:
85 Y=Y+49G-0.4953 R-0.2+80
440 INPUT "BRANCHING NUMBER
850 Y=Z
```

Also change "VERSION 1" to "VERSION 2" in line 40 and "UNIFORM" to "UNIFORMRANDOM" in line 80.

When you run this program without entering new values you will get a uniform tree as before. To introduce the randomness, enter 40 for the branching number, and 80 for the random offset. Of course, a little experimentation is required to get the best values for a realistic tree.

There is still quite a lot that could be added to this program. For example, the branch angle could be made random, or a random factor could decide whether two secondary branches or one would be formed each time. Alternatively, a small routine could be added at the beginning of the program to draw several trees, each with its own branching

Figure 1



Figure 3



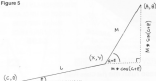
Figure 2



Figure 4



Figure 5



$$\begin{aligned} \text{ie } M &= U\% = (U/5000) - 0.5 \\ \text{and } A &= X + M * \cos(\theta) + 80 \\ B &= Y + M * \sin(\theta) + 80 \end{aligned}$$

Notice: U is the branching ratio.

ching angle, branching number ... and so on.

However, fractals can also be used to good effect in the creation of snowflakes. This time, we begin with an equilateral triangle, then produce an equilateral 'link' on each side, then produce kinks on each side of the first link as in figure six tonight. The program takes the process to the limit of the Dragon's resolution. Unlike the tree program, there is no necessity for a random element to be present, since snowflakes

are perfectly symmetrical. Also, this program is slightly more complicated since more variables have to be dealt with each time, although it behaves in a very similar way. The formula used is derived in figure six. One of the main problems is that the program must know which way up to put the link. The routine at 500 deals with this by looking at which way round the coordinates have been sent to it, then swapping the variables back the right way round and setting a flag, which is later used to determine

Figure 6



Figure 7



Figure 8



the correct angle to use (see Figures ten and eleven). The other main problem is that the Dragon has no APOCDDG command. The correct formula for the inverse-cosine is:

$$1.57 - \text{atan}(\text{Xlog}(\text{X}^2 + 1))$$

You can use this program to generate transverse sections of crystals or crystals by changing the coordinates in line 66, so that a scalene triangle is used. The program can also be modified so that a pentagon is used instead of an equilateral triangle. To do this, simply alter line 70 so that all arrays are dimensioned with 360 instead of 180; replace 3 by 5 in lines 140-160, 210 (where 3/4, 1/4, 1 occur); change the coordinates in line 66 setting 80 to 1/4 and 70 to 3/5 with the required values, and change 105 to 120 to draw five lines instead of three.

The tree and the snowflake crystals are by no means the only structures which are composed of fractals. The coastline of Britain is perhaps the best known example of a

fractal, and in fact it was in trying to answer the question "How long is the coastline of Britain?" that led Mandelbrot to formulate his theory of fractals in 1967. The Dragon can be used to simulate the development of

all sorts of natural structures, from continents to microscopic plants.

More information on fractals can be obtained from *The Fractal Geometry of Nature* by Mandelbrot, published in 1982.

Figure 9



$$\begin{aligned} [A, B] \quad M &= 1/2 * \text{SQR}((G-A)^2 + (H-B)^2) \\ Z &= \text{ARCCOS}((G-A)/M) \\ M &= (Z^2 - 2) \text{ or } (Z^2 + 2) \text{ depending on whether the link is to be facing upwards or downwards} \\ I &= M * \text{COS}(Z) + C \\ J &= M * \text{SIN}(Z) + D \\ C &= 1/2 * (G + 2 * A) \quad D = 1/2 * (H + 2 * B) \quad E = 1/2 * (G + A) \quad F = 1/2 * (H + B) \end{aligned}$$

Figure 10



Figure 11



Program notes

Tree

60-660 Initialise variables, input parameters, clear screen etc.
110 Initialise level counter.
120 Draw the "trunk" of the tree.
130-270 Add branches onto the last primary branch (as shown in X5(Y1)) and store the coordinates of the secondary branches in A(J),B(J) and the new angle in F(J).
280 If level = 7 then the arrays are full,

so wait until a key is pressed before starting again.

240-260 Transfer arrays A(J),B(J) into C(J),D(J).
270-290 Transfer arrays A(J),B(J) into X(J),Y(J).
300-360 Increment level counter then loop back to a further recursive division.
370-380 Routine to draw a line of variable thickness depending on the level. It does not draw the line if any of the coordinates are off the screen.
390-410 Wait routine.
420-480 Routine to get new parameters.

Snowflake

10-70 Initialise dimension arrays.
80 Initialise variables.
90-120 Draw triangle on screen.
130 Initialise level counter.
140-160 Go through each line in previous level, producing kinks and storing new variables in A(J),B(J),C(J),D(J),E(J),F(J),G(J),H(J).
200 Goto routine and wait.

210-240 Get arrays set up for next line round transferring A(J) ... into X(J) ..., Y(J) ...
250-260 Increment level counter then loop back.
270-300 Print snowflake then wait for keypress etc.
310-Routine to produce a short rightway up as in figures ten and eleven.

In both measures M is used as a level indicator. The other variables are as defined in the Harman diagrams, two and nine

TYPE[illegible]

SINOWEI ARE

[illegible]

Dragonsoft

New software for review should be sent to Dragon Guest,
12-13 Little Newport Street, London WC2H 7PP.

It takes two to Kung-Fu

Program: Shaolin Master
Supplier: Quickbeam
Software:
Price: £14.45

THE 68000 of Quickbeam's games revived this issue is a version of the two-player Kung-Fu games so often found in pubs, amusement arcades, etc. The game can be played either one player against a rather deadly computer opponent or two players (one plays each other) which gives this game one good point already.

The game is in two sections, there's the duel between you and an opponent, and then there's the rather gimmicky corridor of death in which you run down a corridor kicking and jumping various large spiky objects.

The game rather like real Kung-Fu needs practice before one can become deadly at it. The controls are straightforward to start with but, like driving a car, after a while they become natural, even second nature. However, to give you an idea of what you're letting yourself in for, the game has the ability to

jump up, high jump, more right, low jump in kneeling position, duck, back somersault, move left, forward somersault, stand up, drop kick, high kick, medium kick, low kick, forward leg sweep, back leg sweep, turn around, high back kick, and low back kick, and all these operations are performed with ONE joystick!

The graphics are as excellent as those of 6800 Express (and that's pretty good) but the game is about thirty times as fast, and thirty times as interesting.

Now for some not picking. The game comes in an attractive cassette box but the instructions are in a separate printed leaflet. Although this seems like a good idea at the time, there should be a reprint of the joystick configurations on the inside of the case along with what each can do, as I managed to lose my sheet three days after receiving the game and thus rendered the game inoperative until I found the two instructions!

Also, the collision detection

routine needs some work on it as it is all too easy for you to lose your figure kick the other guy repeatedly in the groin only to find that you've actually done no damage to him whatsoever! And then, to add insult to injury the device computer assistant will knock you flat with a flying axe kick three inches away from you! I exaggerate, of course, but please, and this goes for all programmers, remember that all we have to

judge by is what we can see on the screen!

To sum up then, a really good game. Let's hope Fire Force is just as good, because it will surely establish Quickbeam—who now have Wayne Southcott working for them, and he is a good programmer—as one of the market leaders.

Jason Chisum



Turtle graphics

Program: LOGO 3.4
Supplier: Bernd Knechtel
Software:, etc.
Price: £16

THE latest piece of Dragon-DOSE software to arrive from Bernd Knechtel is a utility which interfaces directly with the Dragon/Express Basic and provides a few useful additions to the already excellent graphics commands.

LOGO 3.4 provides the user with four sets, graphics commands which allow turtle-type graphics to be programmed. Upon runnings program an invisible 'turtle' is placed at the centre of the screen facing upwards (zero degrees) and this

turtle may be manipulated very easily with the four commands mentioned above. For those who have not had the pleasure of using turtle graphics before, this involves 'guiding' a fictional turtle around the graphics screen with the option of moving, or not leaving, a line drawn in its wake. So it is with a turtle. The first two commands in LOGO 3.4 are TRIGHT and TLEFT which allow you to turn the turtle left or right through some specified angle (up to 90 in the direction you wish to turn).

The position you wish to start from may be defined and re-defined using the TSTART command which sets the initial position of the turtle on the screen. And the line may be drawn (or the turtle moved) using the TMOVE command. The TMOVE command which specifies the distance the turtle is to move (or draw) has an optional parameter which allows a line to be drawn without moving the turtle, or moving the turtle without drawing a line.

Using the example given earlier in this book, the following program would extract the turtle to draw a square on the screen.

```
10 MODE=PCLS SCREEN
11
— Setup graphics mode
20 TSTART 00.00
— Move to centre of the screen
30 TMOVE 50
— Draw a line 50 pixels long
40 TRIGHT 90
— Turn through 90 degrees
50 TMOVE 50
— Draw a line 50 pixels long
60 TRIGHT 90
— Turn through 90 degrees
70 TMOVE 50
— Draw a line 50 pixels long
80 TRIGHT 90
— Turn through 90 degrees
90 TMOVE 50
— Draw a line 50 pixels long
100 END
```

This is obviously a very tedious way of achieving such a simple objective and so, because LOGO interfaces directly with Dragon Basic, a loop may be written which obviously allows for greater flexibility.

For someone who uses Dragon Basic a lot to draw graphics, LOGO 3.4 is a very useful addition to the programmer's armoury. A full assembler listing of LOGO is supplied with the program for anyone daft enough to try and modify it and the package carries no anti-copying devices which makes it easy to incorporate into your own programs. Although LOGO only yields four simple commands in Basic, I can't think of anything else that could be added. LOGO does give the benefits of turtle graphics without the assumption of having to learn an entirely new environment.

Ray Coates



Trouble on the 6809 Express

Program: 6809 Express
Supplier: Quickbeam
Software:
Price: £14.45

OOOOOOOOOOOO!

The runaway game came over the hill and she blew.
The runaway game came over the hill and she blew.

The runaway game came over the hill
And has spent several months on its way down hill
'Cause it's too, too, too boring for you.



The 6809 comes past the trees and it blows.
You play a Mac but that's just the way the game goes.

The programming's all been designed to please

But there's trouble with the strategies.
Unless you, you, you are not a Joe.

The 6809 is a split screen game that goes fast
Below is the top view above are the things as they're passed.

The graphics are great and the sound is hot
It's just a great shame that the game is not.
'Cause it's too, too, too bad to lose track.

Controlling the game requires just as much skill as sleep.
And I think that I've had more fun having a chat with a sheep.

And just 'cause you know how to drive on the screen
Doesn't mean that you've written got a game that's supreme.
So I'm sorry Dave, but this one just won't do.

Jason Chisum



Inside the 32

Dave Barnish strikes a light on the Dragon's memory map

THE average user of the Dragon 32 computer probably sits quite happily at his or her computer, turns it on and starts programming in Basic or using applications programs (or games) written by somebody else, without a lot of concern about how the Dragon really works. Anyone who has dabbled in machine code may sometimes wonder at the complexity of a system that can take a microprocessor, which only understands binary numbers, and produce a computer capable of interacting with the outside world via a keyboard and screen in a language which is not too far removed from English. This article is intended to take a little of the mystery out of the subject and may help those of you feeling confident enough to modify the basic operation of the machine by enhancing the Basic or to use Basic facilities in your own machine code programs.

At the heart of the Dragon is a 6809 microprocessor which is the computer's central processing unit (CPU), where the brainwork goes on. In order for the CPU to do anything it must have instructions, and these are stored in Read Only Memory (ROM). The instructions are built into the ROM when it is made and cannot be altered; they are retained even when the computer is switched off. Random Access Memory (RAM) on the other hand provides a temporary store of data which may be changed at will but is lost when the power is off. In order to communicate with the outside world, some facility for input/output (IO) is required. On the Dragon, the IO is provided by two general purpose IO chips (peripheral interface adapters or PIAs) which, with their associated electronics, provide the keyboard, cassette, sound, printer and joystick interfaces. The screen

is a special case of IO and is handled by a special video display generator chip (VDG).

Because the 6809 is an 8 bit processor, all data is handled in groups of eight bits (8 binary digits) called bytes. For this reason RAM, ROM and IO are split into a number of locations, each of which can hold one byte, and in order that the CPU may distinguish one location from another each is given an individual address which is a number in the range 0 to 65535. The address decoding which organises what goes where is contained in a large chip called a synchronous address multiplexer (SAM), which also does a bit of display management, and maintains the contents of RAM amongst other things. A block diagram of the Dragon's hardware is given in Figure 1, which shows the interconnection between each device. The 6809/SAM and VDG combination is a standard chip set produced by Motorola which has also been adopted by the fairly colour computer. This and the common origin of their Basics explains their internal similarities.

Since the addressing is not based on the decimal system but on binary, it seems reasonable to use decimal ourselves when trying to understand what's going on. However, binary is very unwieldy, so generally the hexadecimal (base 16) system of counting is used which makes things much more manageable. Each hexadecimal (hex) digit has a range of 0 to 15 and to get the numbers 10-15 into one digit, the characters A-F are used. Hexadecimal numbers are in general preceded by a \$ sign, which is a much more conventional way of indicating hex than the \$H that Dragon/Basics uses. Figure 2 gives an example of various numbers expressed in decimal, binary and hexadecimal. Note that each hex digit refers directly to a group of four binary digits (four binary digits make up half a byte which is known as a nibble).

Figure 1 — Hardware block diagram

A = Address bus
D = Data bus

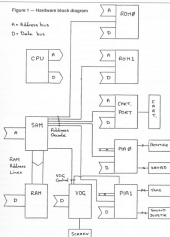


Figure two — Examples of decimal, binary and hexadecimal numbers

Decimal	Binary	Hexadecimal
1	1	1
2	10	2
9	1001	9
10	1010	A
11	1011	B
15	1111	F
16	10000	10
19	10011	13
20	10100	14
31	11111	1F
32	100000	20

A list of which location is assigned to which physical device is known as a memory map, an example of which can be seen in the last page of the Dragon's user manual. This map gives a general idea of where things are, but for a serious investigation of how the system works a more detailed map is required. Such a map is provided in Figure 3 and the rest of this article should be read in conjunction with this map.

In order to appreciate how the Dragon works we had best start at the beginning and find out what happens when the computer is first switched on. When power is

first applied to the machine its electronics come on in an undetermined state, in order to set everything to a known configuration the CPU, IO and SAM chips are provided with a reset input line which is held low (grounded), for a short time after power up by a simple circuit. When the reset line is low the PPIs are cleared and the SAM is set to its initial configuration enabling the address decoding and looking it to the VROM's timing. The CPU then looks at addresses \$FFFF and \$FFFF which SAM maps to ROM locations \$BFFF and \$BFFF; here it finds the reset vector which is the address

of the reset routine to which control is passed. This routine initialises all the I/O, sets up the Basic system variables to their default values, and clears the screen. Having got this far the defined locations of the cartridge memory area are checked (\$C000 and \$C001) and if they contain \$44 and \$48 respectively then control is passed to the cartridge software at locations \$4000. If they two bytes are not set as described the reset routine prints the startup message from locations \$D480 onwards and passes control to the main Basic operating systems.

Figure 3 — Dragon memory map — RAM

\$0000	pointer to beginning of BASIC text	\$00	Printer code, the IO code
\$0010	pointer to beginning of variable space	\$01	Last code read code
\$0020	pointer to beginning of error pointer table	\$02	Printer code
\$0030	initialised to beginning of variable space	\$03	Printer read position
\$0040	low of stack (stack grows down)	\$04-05	Stack Address
\$0050	low of string free space	\$06-09	Start of left and right routine continues to read from a char from BASIC text.
\$0060	Highest Ram available to BASIC	\$0A-0F	Next pointer to current active data
\$0070	pointer to statement to be executed	\$0B-0C	Address of start of I/O routine address table
\$007D	Text pointer for BASIC when start after reset	\$0D	Foreground Colour
\$0080	80th line number	\$0E	Background Colour
\$008D	80th column	\$0F	Active Colour
\$0094	80th pointer		
		\$10	Current Line
\$009D	Current Line Number	\$11-15	Low Address of current window screen
		\$16	Count of lines in a row of graphics
\$00	Current Device Number (PPIs are 0-4)	\$1A-1B	Base address of current graphics screen
\$00	End of this flag (other flag)		
\$00	Restart flag = 0 if 0 455 then 0 else start on reset	\$1C-1D	Current X position
\$0000	Restart Vector = 0 if 0 455 then vector points to a ROM then zero start to ROM after cold start	\$1E-1F	Current Y position
\$00	low address - defined lowest starting		
\$00	low address - defined lowest starting	\$20-21	Current X max, PPIs
\$00	low address - defined lowest starting	\$22-23	Current Y max, PPIs
\$00	low address - defined lowest starting		
\$00	low address - defined lowest starting	\$24	low constant byte
\$00	low address - defined lowest starting	\$25	low constant byte
\$0000	low address - defined lowest starting		
\$00	low address - defined lowest starting	\$26-27	Page No - I/O Drivers - Extension Basic
\$00	low address - defined lowest starting		
\$00	low address - defined lowest starting	\$28-29	Extension vector point to ROM area
\$0000	low address - defined lowest starting	\$2A-2B	\$2A - default 400 00 00
\$00	low address - defined lowest starting	\$2C-2D	\$2C - default 400 00 00
\$00	low address - defined lowest starting	\$2E-2F	\$2E - default 400 00 00
\$0000	low address - defined lowest starting	\$30-31	\$30 - default 400 00 00
\$00	low address - defined lowest starting	\$32-33	\$32 - default 400 00 00
\$0000	low address - defined lowest starting	\$34-35	\$34 - default 400 00 00
\$00	low address - defined lowest starting	\$36-37	\$36 - default 400 00 00
\$0000	low address - defined lowest starting	\$38-39	\$38 - default 400 00 00
\$00	low address - defined lowest starting	\$3A-3B	\$3A - default 400 00 00
\$0000	low address - defined lowest starting	\$3C-3D	\$3C - default 400 00 00
\$00	low address - defined lowest starting	\$3E-3F	\$3E - default 400 00 00
\$0000	low address - defined lowest starting	\$40-41	\$40 - default 400 00 00
\$00	low address - defined lowest starting	\$42-43	\$42 - default 400 00 00
\$0000	low address - defined lowest starting	\$44-45	\$44 - default 400 00 00
\$00	low address - defined lowest starting	\$46-47	\$46 - default 400 00 00
\$0000	low address - defined lowest starting	\$48-49	\$48 - default 400 00 00
\$00	low address - defined lowest starting	\$4A-4B	\$4A - default 400 00 00
\$0000	low address - defined lowest starting	\$4C-4D	\$4C - default 400 00 00
\$00	low address - defined lowest starting	\$4E-4F	\$4E - default 400 00 00
\$0000	low address - defined lowest starting	\$50-51	\$50 - default 400 00 00
\$00	low address - defined lowest starting	\$52-53	\$52 - default 400 00 00
\$0000	low address - defined lowest starting	\$54-55	\$54 - default 400 00 00
\$00	low address - defined lowest starting	\$56-57	\$56 - default 400 00 00
\$0000	low address - defined lowest starting	\$58-59	\$58 - default 400 00 00
\$00	low address - defined lowest starting	\$5A-5B	\$5A - default 400 00 00
\$0000	low address - defined lowest starting	\$5C-5D	\$5C - default 400 00 00
\$00	low address - defined lowest starting	\$5E-5F	\$5E - default 400 00 00
\$0000	low address - defined lowest starting	\$60-61	\$60 - default 400 00 00
\$00	low address - defined lowest starting	\$62-63	\$62 - default 400 00 00
\$0000	low address - defined lowest starting	\$64-65	\$64 - default 400 00 00
\$00	low address - defined lowest starting	\$66-67	\$66 - default 400 00 00
\$0000	low address - defined lowest starting	\$68-69	\$68 - default 400 00 00
\$00	low address - defined lowest starting	\$6A-6B	\$6A - default 400 00 00
\$0000	low address - defined lowest starting	\$6C-6D	\$6C - default 400 00 00
\$00	low address - defined lowest starting	\$6E-6F	\$6E - default 400 00 00
\$0000	low address - defined lowest starting	\$70-71	\$70 - default 400 00 00
\$00	low address - defined lowest starting	\$72-73	\$72 - default 400 00 00
\$0000	low address - defined lowest starting	\$74-75	\$74 - default 400 00 00
\$00	low address - defined lowest starting	\$76-77	\$76 - default 400 00 00
\$0000	low address - defined lowest starting	\$78-79	\$78 - default 400 00 00
\$00	low address - defined lowest starting	\$7A-7B	\$7A - default 400 00 00
\$0000	low address - defined lowest starting	\$7C-7D	\$7C - default 400 00 00
\$00	low address - defined lowest starting	\$7E-7F	\$7E - default 400 00 00
\$0000	low address - defined lowest starting	\$80-81	\$80 - default 400 00 00
\$00	low address - defined lowest starting	\$82-83	\$82 - default 400 00 00
\$0000	low address - defined lowest starting	\$84-85	\$84 - default 400 00 00
\$00	low address - defined lowest starting	\$86-87	\$86 - default 400 00 00
\$0000	low address - defined lowest starting	\$88-89	\$88 - default 400 00 00
\$00	low address - defined lowest starting	\$8A-8B	\$8A - default 400 00 00
\$0000	low address - defined lowest starting	\$8C-8D	\$8C - default 400 00 00
\$00	low address - defined lowest starting	\$8E-8F	\$8E - default 400 00 00
\$0000	low address - defined lowest starting	\$90-91	\$90 - default 400 00 00
\$00	low address - defined lowest starting	\$92-93	\$92 - default 400 00 00
\$0000	low address - defined lowest starting	\$94-95	\$94 - default 400 00 00
\$00	low address - defined lowest starting	\$96-97	\$96 - default 400 00 00
\$0000	low address - defined lowest starting	\$98-99	\$98 - default 400 00 00
\$00	low address - defined lowest starting	\$9A-9B	\$9A - default 400 00 00
\$0000	low address - defined lowest starting	\$9C-9D	\$9C - default 400 00 00
\$00	low address - defined lowest starting	\$9E-9F	\$9E - default 400 00 00
\$0000	low address - defined lowest starting	\$A0-A1	\$A0 - default 400 00 00
\$00	low address - defined lowest starting	\$A2-A3	\$A2 - default 400 00 00
\$0000	low address - defined lowest starting	\$A4-A5	\$A4 - default 400 00 00
\$00	low address - defined lowest starting	\$A6-A7	\$A6 - default 400 00 00
\$0000	low address - defined lowest starting	\$A8-A9	\$A8 - default 400 00 00
\$00	low address - defined lowest starting	\$AA-AB	\$AA - default 400 00 00
\$0000	low address - defined lowest starting	\$AC-AD	\$AC - default 400 00 00
\$00	low address - defined lowest starting	\$AE-AF	\$AE - default 400 00 00
\$0000	low address - defined lowest starting	\$B0-B1	\$B0 - default 400 00 00
\$00	low address - defined lowest starting	\$B2-B3	\$B2 - default 400 00 00
\$0000	low address - defined lowest starting	\$B4-B5	\$B4 - default 400 00 00
\$00	low address - defined lowest starting	\$B6-B7	\$B6 - default 400 00 00
\$0000	low address - defined lowest starting	\$B8-B9	\$B8 - default 400 00 00
\$00	low address - defined lowest starting	\$BA-BB	\$BA - default 400 00 00
\$0000	low address - defined lowest starting	\$BC-BD	\$BC - default 400 00 00
\$00	low address - defined lowest starting	\$BE-BF	\$BE - default 400 00 00
\$0000	low address - defined lowest starting	\$C0-C1	\$C0 - default 400 00 00
\$00	low address - defined lowest starting	\$C2-C3	\$C2 - default 400 00 00
\$0000	low address - defined lowest starting	\$C4-C5	\$C4 - default 400 00 00
\$00	low address - defined lowest starting	\$C6-C7	\$C6 - default 400 00 00
\$0000	low address - defined lowest starting	\$C8-C9	\$C8 - default 400 00 00
\$00	low address - defined lowest starting	\$CA-CB	\$CA - default 400 00 00
\$0000	low address - defined lowest starting	\$CC-CD	\$CC - default 400 00 00
\$00	low address - defined lowest starting	\$CE-CF	\$CE - default 400 00 00
\$0000	low address - defined lowest starting	\$D0-D1	\$D0 - default 400 00 00
\$00	low address - defined lowest starting	\$D2-D3	\$D2 - default 400 00 00
\$0000	low address - defined lowest starting	\$D4-D5	\$D4 - default 400 00 00
\$00	low address - defined lowest starting	\$D6-D7	\$D6 - default 400 00 00
\$0000	low address - defined lowest starting	\$D8-D9	\$D8 - default 400 00 00
\$00	low address - defined lowest starting	\$DA-DB	\$DA - default 400 00 00
\$0000	low address - defined lowest starting	\$DC-DD	\$DC - default 400 00 00
\$00	low address - defined lowest starting	\$DE-DF	\$DE - default 400 00 00
\$0000	low address - defined lowest starting	\$E0-E1	\$E0 - default 400 00 00
\$00	low address - defined lowest starting	\$E2-E3	\$E2 - default 400 00 00
\$0000	low address - defined lowest starting	\$E4-E5	\$E4 - default 400 00 00
\$00	low address - defined lowest starting	\$E6-E7	\$E6 - default 400 00 00
\$0000	low address - defined lowest starting	\$E8-E9	\$E8 - default 400 00 00
\$00	low address - defined lowest starting	\$EA-EB	\$EA - default 400 00 00
\$0000	low address - defined lowest starting	\$EC-ED	\$EC - default 400 00 00
\$00	low address - defined lowest starting	\$EE-EF	\$EE - default 400 00 00
\$0000	low address - defined lowest starting	\$F0-F1	\$F0 - default 400 00 00
\$00	low address - defined lowest starting	\$F2-F3	\$F2 - default 400 00 00
\$0000	low address - defined lowest starting	\$F4-F5	\$F4 - default 400 00 00
\$00	low address - defined lowest starting	\$F6-F7	\$F6 - default 400 00 00
\$0000	low address - defined lowest starting	\$F8-F9	\$F8 - default 400 00 00
\$00	low address - defined lowest starting	\$FA-FB	\$FA - default 400 00 00
\$0000	low address - defined lowest starting	\$FC-FD	\$FC - default 400 00 00
\$00	low address - defined lowest starting	\$FE-FF	\$FE - default 400 00 00

[illegible]

Figure 3 (cont'd) — (Source: memory map, — IBM — 1980, 1981)

[illegible]

[illegible]

autonomous service routine, thus giving you a permanent clock on screen without interfering with Basic. Figure 4 is a listing of a machine code program which does this.

The locations assigned to ROM, RAM and I/O are fixed by the address decoding of SAM but the use to which RAM and I/O is put is determined by the CPU instructions stored in ROM. The actual functioning of the RAM has already been discussed in previous editions of Dragon Users. July 1983 had an article with a few errors which were corrected in March 1984 (so the data has been omitted from the map at Figure 3; however, the usage of RAM gives a good insight into the way Basic actually works and is well worth some investigation).

The area from \$00 to \$FF is known as page zero and is used by the system to store various system variables. The start and end addresses of any program entered are stored here as well as pointers to the variable storage area and many other pieces of information which collectively decide the current state of Basic. Useful locations are \$10 and \$14, which point to the beginning of Basic text programs can be merged by manipulating these locations as follows:

1) Turn the Dragon off and on to make sure the pointers are set to their default values.
2) Load the first program — this will be stored at \$1000 which is pointed to by \$1014 as a default.

3) Change the pointer as \$1014 to point to the space after the program. The space after the program is the simple variable space pointed to by \$1016; so \$1014 should be given this value minus two, to get over the last two bytes of the first program, which are delimiters (see later).

4) Load the second program and remember it so that the line numbers are higher than those in the first program.

5) Restore the start of program pointer to its original value (\$1000).

The process is accomplished from Basic thus:

CLDMD "first prog" — first program is loaded at \$1000.
POKE \$10,PEEK(\$?) — change pointer to end
POKE \$10,PEEK(\$?) + 2 — of program.

CLDMD "second prog" — second program is loaded after first.

RENUM 1,2 — change conflicting line numbers.
POKE \$5,30 — return pointer to beginning
POKE \$5,1 — of first program.

Locations \$03 and \$04 point to the next data statement to read and can be used to provide a kind of random access data statement. Locations \$70 and \$71 hold the wait start sector: when the reset button on the side of the Dragon is pushed location \$71 is checked and if it contains \$FF and location \$70 and \$71 points to a NOP/NoOperation — code \$105, then control passes to that NOP which normally results in the screen being cleared and the message 'OK' being printed. This vector can be changed to point to your own reset routine, a technique used in *Mastermind's* The King amongst others.

The system also uses the area \$400 to \$7FF which contains Basic's statement dispatch table (more about this later), the

lookup table, the joystick readings and a very useful set of patch routines into Basic itself. These patches are three byte subroutines which are called when Basic is executing certain commands. By default these subroutines contain \$06 which is the code for RTS (Return from Subroutine), so they don't do anything at all. However, because there are three bytes to each patch, there is room to insert a jump instruction so that when the subroutine is called, it jumps to your own routine which can do something other than returning. An example of this is intercepting the patch into the LIST command in \$4A8 with a delaying routine to slow down listings (see Dragon User October 1983 page 11). It is also possible to disable the break key by intercepting the read in next statement patch in \$7E4 and returning with the return address incremented by four, which bypasses the break key check.

The final area of RAM reserved for the system is from \$800 to \$FFF which is principally used for the cassette buffer.

The next several locations hold the information which will be displayed on the screen. The area from \$A00 to \$FFF is the default location for the text screen (this can be changed as in the Dragon Toolkit from Premier, which gives 25 different text screens). The text screen address is put in \$A00 which tells the VDG which area of memory it is to read from and what mode it is in. In text mode the data stored in each byte of the screen memory is used to access a character generator which is built into the VDG, and which provides the dot patterns which make up each character. The screen code for each character is given in the back of the Dragon manual.

After the text screen, the graphics pages start (\$800 onwards). The amount of memory used for graphics depends on how many pages have been POLEARed, the default being four pages, which uses up the area from \$800 to \$1FFF. If eight pages are used, the graphics memory extends to \$3FFF leaving 16K of RAM for your Basic programs and variables, which are stored immediately after the graphics pages up to location \$FFFF. If you change the number of graphics pages while you have a program in memory, the system automatically moves the program up or down in memory to follow in PMODES, each byte in the screen data represents four dots, the two bits for each dot allow the four possible colours to be encoded. PMODE4 represents each dot on the screen by one bit. As a bit can be either 1 or 0 there are only two colours available in this mode.

After the user RAM area the address space is given over to ROM which resides from location \$8000 to \$FFFF. This ROM contains the instructions that enable the Dragon to understand Basic; what follows is a rough outline of how this is accomplished.

When a program is typed in, it is stored in the user RAM area according to the start and end of text pointers in zero page. The program is stored as shown:

Storage format of Basic text:

1 byte containing 0 in the first program location.

A number of Basic lines stored as:
2 bytes to point to the start location of the

next line.

2 bytes for the line number (in hex).

Up to 248 bytes for the line text.

1 byte containing 0 to mark the end of the line.

And
2 bytes containing zero to mark the end of the program.

Each character which the machine can display is represented in memory by an eight bit number — it's own ASCII code (ASCII stands for the American Standard Code for Information Interchange). The ASCII code is a standard system adopted by most machines (the Sinclair computers being one notable exception) and is listed in Figure 5. The reserved words, such as PRINT and READ, are not stored in their ASCII format but are converted to a one byte token, thus cutting down on storage used by a great deal. Functions, such as LEN or ASL, are represented by a token preceded by \$FF. Figure 6 contains a list of reserved words and their respective tokens.

When the LIST command is used the tokens encountered in each line are looked up in a table and the full word printed out. Converting reserved words to tokens is often known as 'tokenizing' and tokenizing is the crux of the routines which do this are the \$10F08 and \$10F0E. As an example of using Basic's routines from your own programs, Figure 6 contains a program which produces nearly formatted listings as a primer using the do-crunch routine from Basic. Looking through the reserved word table turns up a word which isn't explained in the manual. The word is DLOAD and if you enter it into your Dragon, you will get an I/O error. It looks like DLOAD is a relic from Sandy colour Basic, where the Dragon Basic has its roots, which has an \$E030 serial I/O port input command — DLOAD.

When a Basic program is RUN, tokens are used to point into the statement dispatch table. This table contains the address of the routine which executes the reserved word designated by the token; looking through the memory map shows where some of these routines are to be found.

The area of RAM between the end of the Basic program and address \$FFFF is available for the storage of Basic variables. There are four types of variables in Dragon Basic: simple numbers, simple string, numeric arrays and string arrays and each is stored in a particular format as shown in Figure 7. Simple variables are stored immediately after the Basic program and in variables are stored after these. If a new simple variable is encountered, all the arrays which have been defined are moved up to make room, this process being known as dynamic memory allocation. String variables of both types are stored as pointers to the actual location where the string is stored. If the string is defined in the program text, the pointer will point there, if not, the area at the top end of memory is reserved for strings, and the pointer will point to that area. Pointers in zero page are used to keep track of the start and end addresses of the simple variables, array variables and string storage space. See locations \$10 to \$20 on the memory map.

Under 4-11 are several paragraphs
which are removed and are not to be used for anything in House

[illegible]

Method used	Dataset	Assessment	Platform	Assessment	Version	Assessment	Task type
(continued)		(continued)		(continued)			

[illegible]

1. *Journal of the American Medical Association*, 1997; 278: 1039-1044.

• A byte containing the binary number 100100101011011.

1. $\text{Total} = \text{Total}$ containing the number of elements in the array.

² Further contributions: *Journal of the Philosophy of Education Society of Great Britain*, 35 (2004), 1–20.

The string itself is stored in ASCII at the top of memory.

2) Optimal scheduling that allocates codes for that allocation scheme.

1 byte containing the number of

8 bytes per element of the array. Each element

As a result, the model is able to capture the underlying structure of the data, and the results are more reliable and accurate.

if (type == 0) // not a variable name -- ignored!
 type = type > 0 ? 0 : 1;

2 bytes per dimension containing the

Bytes per element of the array. Each ele-

name is stored in the same way as a simple string without the name.

There is a more detailed explanation on programs and variable storage in this March.

[illegible][illegible]

The TRS80 Colour Computer Technical Reference Manual from Sandy. This contains a lot of information relevant to the Dragon.

Inside The Dragon by Christine Seward and Heri Formanville

Memory Browser

This short program allows one to browse through the memory of the Dragon. After entering the start address, which can be in either in hex or decimal, the contents of 66 bytes from that address are displayed, together with the equivalent ASCII character. Any non-ASCII or control code characters are displayed as full stops. The user can then browse backwards or forwards through the memory by using the up

and down arrows, or restart from a new address by touching the spacebar. Interesting sample browsers start at hex 8000 (Basic program area, unless you have PCLEAR), and hex 8000 (the BASIC keywords).

The superiority of Dragon Microsoft BASIC was simply demonstrated when I tried to adapt this program for the Commodore 64, so PRINT USING, no HEX\$, no PRINT% etc made it fail as long again!

In retrospect, it's a pity that Dragon Data didn't make better play of the commonality of Dragon Basic with that supplied to IBM for the PC and XT, both highly respected machines.

*Hugh Pattfield
Room 206
Billeted House
Woburn, Bedford
London NW1 6SL*

```
100 A$=CHR$(120):S$=STRING$(32,A$):REM Memory Display by H Pattfield
110 B$=A$+"addr"+"00"+A$+"01"+A$+"02"+A$+"03"+A$+"04"+A$+"05"+A$
+A$+"012345"+A$+REM TYPE addr IN REVERSE VIDEO
120 C$=A$+"% 3"+A$+"%2"+A$+"%2"+A$+"%3"+A$+"%3"+A$+"%2"+A$+"%2"+A$
+A$+"% 3"+A$
130 B$=A$+"hex"+STRING$(32,A$)+"ascii":REM TYPE hex AND ascii IN RE
VERSE VIDEO
140 CLS:INPUT "START ADDRESS>";A
150 CLSO:PRINT B$;D$;S$;
160 PRINT%452,"USE UP & DOWN ARROWS,";
170 PRINT%454,"OR SPACEBAR TO RESTART";
180 PRINT%55,"";
190 FOR ROW=0 TO 9
200   B$=HEX$(A+(ROW*6))
210   IF LEN(B$)<4 THEN B$="0"+B$:GOTO 210
220   P$="....."
230   FOR COL=0 TO 5
240     FICOL=PEEK(A+(ROW*6)+COL)
250     IF (FICOL)>32 OR FICOL<127 THEN 270
260     HIB$(F$(COL+1,1))=CHR$(FICOL)
270     GB$(COL)=HEX$(FICOL)
280     IF LEN(GB$(COL))=1 THEN GB$(COL)="0"+GB$(COL)
290   NEXT COL
300   PRINT "=-Z,USING C$;B$,GB$(1),GB$(1),GB$(2),GB$(3),GB$(4),GB$(5),P$;
310 NEXT ROW
320 P$=INKEY:IF P$="" THEN 320
330 IF ASC(P$)=94 THEN A=A+65:GOTO 180
340 IF ASC(P$)=10 THEN A=A+45:GOTO 180
350 IF ASC(P$)=32 THEN 140
360 CLS:STOP
```

Intelligent Pattern Generator

THIS has been written in response to all those letters asking for short programs. It will fill the screen with a random pattern, and is held viewed on a colour TV or monitor.

Lines 1 and 2 set random co-ordinates; lines 3 and 4 work out where to put the patterns; line 5 and 6 draw the patterns to the screen; lines 7 and 8 move through the various screens.

*A. Bailey (Gann)
32 Pinewood St
Chislehurst
Suffolk SS27 6JN*

```
1 P=ABS(SCREEN/5):Q=ABS(SCREEN/5)+1:R=ABS(SCREEN/5)+1
2 A=ABS(50)+B=ABS(40)+1:D=ABS(40)+1
3 B=ABS(100):C=INT(ABS(100)+B)/2:A=ABS(100)/2
4 Q=ABS(100):D=INT(ABS(100)+Q)/2+B=ABS(100)/2
5 FOR X=0 TO 255:FOR Y=0 STEP A:FOR Z=0 TO 190:FOR W=0 STEP B
6   CIRCLE(X,Y),C,C7NEXT Y,X:FOR W=0 TO 4
7   FOR J=0 TO 1:P=ABS H:SCREEN J,1:FOR Z=0 TO 255
8   NEXT Z,J,H:GOTO 5B=3:GOTO 1
```


Expert's Arcade Arena

Write to "The Expert" at Dragon User
13-13 Little Newport St, London WC2H 9PP,
with all your arcade tips and hints.

AND so the new year dawns, bringing with it new hope (unless you own a Dragon, in which case you've probably completely given up hope!), new life, new civilisations, lovely going where to than has, and soon it'll appear to have lost track time a bit.

Anyway, the new year heralds something else, the new year brings another TWELVE MONTHS of the award winning EXPERT'S ARCADE ARENA, so we're now again, last night at the prestigious AFCEAAA awards ("Awards For Columns Called Experts Arcade Arena") which took place in my bedroom, this column took every award available. Yes, you are reading the "Best Monthly Column in Dragon User Called Experts Arcade Arena", the "Most Witty Column Called Expert's Arcade Arena", and last but by no means least the "Most Elegant And Clever Column With The Most Awards Ever In A Magazine Called Dragon User".

So, a new year, but still the same old jokes. As I'm writing this column in November and it's not really the new year yet, I haven't received your Christmas gifts yet, but a big thank you in advance for them and the winner of the "Really Rather Silly Present" competition will be announced soon!

Now then, if you remember a few months back I published a complete list of the messages at the end of each screen on Time Bandit and asked you to send me a complete list. Now, I wasn't really that convinced that anyone would be stupid enough to copy out a list and post it into the postbox or me. I copy out nearly everything you write and post it to the entire readership — *ZZ!* but I thought it was worth a try for a good laugh and YES! Someone actually bothered to do a competition with no question SQ, I am proudly announce that Philip Saffer received the award for "Possibly the most stupid person in the world who owns a Dragon and reads Expert's Arcade Arena and has nothing better to do than copy out lines of text from a column called Expert's Arcade Arena in a magazine called Dragon User" (Yes, this sounds very familiar — *ZZ!*).

Philip has also sent some pointers to help those among you who aren't smart enough to play games by yourselves ...

DRAGON HUNK: "Load the game with the following command: CLOADM 115606: (POKE 2084,17) EXEC 50158 then when the game has loaded type POKE 1598,157: (EXEC 5006)

... unfortunately all of the rest of Philip's pointers were covered by me in my last column, so tough titty there Philip! Happy Christmas!

ORMM: Now then, on the subject of last month's issue, if you'd affix it off and another on page 4 in the bottom left hand corner you'll see an apology, now then, the last time I wrote "Dragon User and Mr Barclay was, let me apologise to all concerned for the misunderstanding." You notice it (DOESN'T



HEY "Dragon User and Mr Barclay and The Expert would like to apologise to all concerned for this misunderstanding." Well, in case you were wondering, I'll tell you it's because I'm NOT SORRY AT ALL! In fact, I find it RATHER FUNNY! However, I would like to apologise on behalf of my editor (and I say "my" because I do in fact own her!) for the fact that a map of the game Fantasy Flight was called Fantasy Flight, the Map! (Please refer above for explanation. And be aware: Care! — *ZZ!*). I can assure my loyal readership that Mr. Armstrong has received 40 letters as well as being told ... sorry, get off the back again there.

Some of you may be wondering "Where is the map of this month's great one?" Well, the truth is, there ISN'T ONE, so you can stop looking for it. Instead, next month's Arcade Arena will be a MAP SPECIAL, with maps of Knight's Kingdom, stages two and three of Castle Smash (with four and five to follow in March), Templewood and a rather incomprehensible map of Universe One to boot! So please, I warn you in advance that this one is going to be difficult to read, but I've decided to publish it because it's so damn useful!

Now then, to Joe Brimcat from Malta who really sent probably the best written letter I've ever seen. (Which sounds mad, I must say) (my the rare competition) and he asks why I WRITE and SAUCE or later (Colours are always out of touch. The answer is simple because those options are not available until later Universes.

Now then, to the competition ... we are still running the following competitions most of which are free to enter and have temporarily non-existent prizes:

What is the best music to play to? (Latest nomination is Keith Jarrett, by me)

What are the codes at the end of each screen on Time Bandit? (well, it's got to be worth a try, hasn't it?)

Who can provide the silent Christmas present to the Expert?

Who is the Expert, and who does he or she have such strange musical tastes?

AND NOW ... THE NEW COMPETITION...

Who can send the best written letter to the Expert? (I mean it's handwriting, not use of English)

Right, now to move on to a letter from M. "Mr, I'm Not A Software Please Really 'I'm Honour' Vins, who asks for more maps (your place is answered) and provides more jokes (them, strange contradiction somewhere there) so that those of you who don't want to play in black and white can move up to glorious full colour. The addresses are:

ACQUAINT 471: POKE 15306,204
THEARCHER: POKE 7340,204 (or 202)
VORTEX FACTOR: POKE 7063, 204
SHENANIGANS: POKE 7063, 204
ZAXXON (from 58504): POKE 26320, 204 (or 202)

To use these codes you need PEARL 2. Dragon's leader programs which have taken the course of other phone calls and letters from any other, and for the first time (and only time until January 1988) fees are in the famous programs A, B and C:

PROGRAM A:

10 A=108: POKE A,142: POKE A+1,4:
POKE A+2,0: POKE A+3,126: POKE
A+4,185: POKE A+5,51: STOP:
EXEC A

PROGRAM B:

10 A=30850: POKE A,185: POKE A+
1,160: POKE A+2,146: POKE A+3,
134: POKE A+4,57: POKE A+5,183:
POKE A+6,1: POKE A+7,183: POKE
A+8,57: EXEC A

PROGRAM C:

10 PCLR=0: CLEAR 16: FOR I=1080
TO 1661: READ M: POKE I,M: NEXT
EXEC 1663:30 EXEC PEEK
15487:208 + PEEK(1547)
30 DATA 154, 155, 163, 1, 142, 48, 142,
26, 161, 1, 143, 141, 31, 141, 34, 134, 3,
151, 0, 142, 84, 0, 141, 8, 10, 0, 36, 247,
160, 84, 2, 155, 0, 126, 183, 81, 168, 0,
141, 2, 141, 2, 52, 144, 142, 8, 0, 168,
126, 165, 165, 50

Right, that's it for this month, keep the letters coming, see you soon,
etc...etc...etc...

Functions

I have had problems converting all the functions to ACS and ASA (Ac Collins and Art Gies). Those in the manual appear to be incorrect; please could you give the correct versions.

J. F. Joneswood
81 Haydon Road
Milton Keynes

YOU can write the following function definitions to return the art size and art columns respectively:

```
DEF FNS(3)=ART(3/32)*X*Y+10
DEF
FNS(2)=ART(3/32)*Y*2+13
```

Restore

I have present writing a game for my Dragon 32. However I have come across a problem. In a DOS statement I need to restore pieces of information from a DOS statement at random. If there was a RESTORE in command I would help tremendously.

David May
67 Haydon Road
Milton Keynes

ALTHOUGH Dragon Basic does include the RESTORE command there is no facility to restore an individual line. I gave a short routine to do this in a recent DU. Alternatively read all your data into an array at the start of the game, and access it directly in the array. This will generally be faster than using RESTORE. A hint if you never repeatedly use READ/DATA: put all the DATA statements at the start of the program.

Not ready

I AM writing a wordprocessor program for my Dragon 4 with Dragon 32s, but I have run into a problem. How does the Ready and Write Protect lines of the drive to see if there is a disc present in the drive.

The Dragon 32 routines just look at the disc's presence, so how can I proceed then?

Sam Edwards
Salford
Peter Oldlands

UNFORTUNATELY, Dragon32 does not support the 'ready' line from the disc drive, and so this can not be used to check if the drive is ready. However, it is possible to detect if a disc is in the drive and ready by inspecting the index strobe input.



Initially the DOS routine at 49900 should be called, this returns the drive status in the command register at 5F448. If bit 6 is set then the disc is write protected. If bit 2 is zero then the drive is not on line and is hence not ready. Bit 1 in the index input, this goes high when the index data passes under the led in the drive. Your machine code routine should look at this in a loop (about 5000 times say) and until it changes — if it does then a disc is in the drive and the disk is closed. This should be done with interrupts disabled.

Link up

I have a Dragon 64 and two printers, one with serial interface. I have been trying for sometime to link up the serial printer to the Dragons 64000 port but have been unsuccessful.

I have tried connecting the following lines, GND to GND (GND to GND and GND to GND). The problem is that the Dragon just freezes up whenever I send a character to the printer. Examining the value at 407F0 seems to indicate that the printer is permanently busy?

Ken Thomas
Salford
Neil Millward

THIS is quite a common problem on which I regularly get letters. The problem is that the output 'status' from the control on many printers goes 'high' (+5V) when the printer is busy and 'low' when ready. The Dragon's 'CTS' expects the opposite to this, is high when ready, low when busy.

Unfortunately, there is no simple solution to this. You could build a small interface consisting of an hex inverter chip to invert the status

output. Alternatively you can try increasing the end-of-line delay value at 507F0/003 and being 'CTS' to +5V permanently.

Printer

THE PROBLEM is that I have decided to get a printer for my Dragon 32. However, as you can understand, the problem of which one to choose is extremely difficult.

One machine that certainly may attract others at others, this being the CMP-2000 from Amstrad, it is within the price range I'm looking at and has a near letter quality print which I need.

With this in mind my question is quite simple. Will my computer be able to use this printer, or another two machines simultaneously.

George Cooke
54, Westwood Lane
Sarnborough

THE AMSTRAD CMP-2000 has a standard 8-bit Centronics parallel interface and will work quite happily with a Dragon computer. In fact, because of the Dragon's 8-bit interface, it is possible to directly access the extended channels available on the printer, which can not be done from an Amstrad more (which have 1-bit interfaces).

You will need a suitable printer lead, any Dragon lead should work, but those supplied with the printer will only fit Amstrads.

Pins

AT SCHOOL I am about to do a Computer Studies Project. I would like to do my project on my Dragon 32 as it

is much better than the school computers, but I do not have a printer. My teacher could put together a makeshift printer lead to use with a serial printer, but I need to know the separate pin connections for the Dragon's printer port.

Could you give me this information, or tell me where I can obtain it from?

R. Boucher
1 Elmwood Road
Sarnborough

THE pin connections for the printer port can be found at the back of most Dragon manuals, or in the separate additional information booklet. However, this question does come up quite regularly so I'll repeat it here.

- Odd numbered pins on top line, pins 1 on the right (printed and not), even pins on bottom, pin 2 on the right again.
- PIN 1 Print Stroke PIN 2 +5 volts
 - 3 Data bit 0-1 +5 volts
 - 4 Data bit 1-0 Ground
 - 5 Data bit 2-0 Ground
 - 6 Data bit 3-0 Ground
 - 7 Data bit 4-10 Ground
 - 8 Data bit 5-10 Ground
 - 9 Data bit 6-10 Ground
 - 10 Data bit 7-10 Ground
 - 11 Data bit 8-10 Ground
 - 12 Data bit 9-10 Ground
 - 13 Data bit 10-10 Ground
 - 14 Data bit 11-10 Ground
 - 15 Data bit 12-10 Ground
 - 16 Data bit 13-10 Ground
 - 17 Data bit 14-10 Ground
 - 18 Data bit 15-10 Ground
 - 19 Data bit 16-10 Ground
 - 20 Data bit 17-10 Ground
 - 21 Data bit 18-10 Ground
 - 22 Data bit 19-10 Ground
 - 23 Data bit 20-10 Ground
 - 24 Data bit 21-10 Ground
 - 25 Data bit 22-10 Ground
 - 26 Data bit 23-10 Ground
 - 27 Data bit 24-10 Ground
 - 28 Data bit 25-10 Ground
 - 29 Data bit 26-10 Ground
 - 30 Data bit 27-10 Ground
 - 31 Data bit 28-10 Ground
 - 32 Data bit 29-10 Ground
 - 33 Data bit 30-10 Ground
 - 34 Data bit 31-10 Ground
 - 35 Data bit 32-10 Ground
 - 36 Data bit 33-10 Ground
 - 37 Data bit 34-10 Ground
 - 38 Data bit 35-10 Ground
 - 39 Data bit 36-10 Ground
 - 40 Data bit 37-10 Ground
 - 41 Data bit 38-10 Ground
 - 42 Data bit 39-10 Ground
 - 43 Data bit 40-10 Ground
 - 44 Data bit 41-10 Ground
 - 45 Data bit 42-10 Ground
 - 46 Data bit 43-10 Ground
 - 47 Data bit 44-10 Ground
 - 48 Data bit 45-10 Ground
 - 49 Data bit 46-10 Ground
 - 50 Data bit 47-10 Ground
 - 51 Data bit 48-10 Ground
 - 52 Data bit 49-10 Ground
 - 53 Data bit 50-10 Ground
 - 54 Data bit 51-10 Ground
 - 55 Data bit 52-10 Ground
 - 56 Data bit 53-10 Ground
 - 57 Data bit 54-10 Ground
 - 58 Data bit 55-10 Ground
 - 59 Data bit 56-10 Ground
 - 60 Data bit 57-10 Ground
 - 61 Data bit 58-10 Ground
 - 62 Data bit 59-10 Ground
 - 63 Data bit 60-10 Ground
 - 64 Data bit 61-10 Ground
 - 65 Data bit 62-10 Ground
 - 66 Data bit 63-10 Ground
 - 67 Data bit 64-10 Ground
 - 68 Data bit 65-10 Ground
 - 69 Data bit 66-10 Ground
 - 70 Data bit 67-10 Ground
 - 71 Data bit 68-10 Ground
 - 72 Data bit 69-10 Ground
 - 73 Data bit 70-10 Ground
 - 74 Data bit 71-10 Ground
 - 75 Data bit 72-10 Ground
 - 76 Data bit 73-10 Ground
 - 77 Data bit 74-10 Ground
 - 78 Data bit 75-10 Ground
 - 79 Data bit 76-10 Ground
 - 80 Data bit 77-10 Ground
 - 81 Data bit 78-10 Ground
 - 82 Data bit 79-10 Ground
 - 83 Data bit 80-10 Ground
 - 84 Data bit 81-10 Ground
 - 85 Data bit 82-10 Ground
 - 86 Data bit 83-10 Ground
 - 87 Data bit 84-10 Ground
 - 88 Data bit 85-10 Ground
 - 89 Data bit 86-10 Ground
 - 90 Data bit 87-10 Ground
 - 91 Data bit 88-10 Ground
 - 92 Data bit 89-10 Ground
 - 93 Data bit 90-10 Ground
 - 94 Data bit 91-10 Ground
 - 95 Data bit 92-10 Ground
 - 96 Data bit 93-10 Ground
 - 97 Data bit 94-10 Ground
 - 98 Data bit 95-10 Ground
 - 99 Data bit 96-10 Ground
 - 100 Data bit 97-10 Ground
 - 101 Data bit 98-10 Ground
 - 102 Data bit 99-10 Ground
 - 103 Data bit 100-10 Ground
 - 104 Data bit 101-10 Ground
 - 105 Data bit 102-10 Ground
 - 106 Data bit 103-10 Ground
 - 107 Data bit 104-10 Ground
 - 108 Data bit 105-10 Ground
 - 109 Data bit 106-10 Ground
 - 110 Data bit 107-10 Ground
 - 111 Data bit 108-10 Ground
 - 112 Data bit 109-10 Ground
 - 113 Data bit 110-10 Ground
 - 114 Data bit 111-10 Ground
 - 115 Data bit 112-10 Ground
 - 116 Data bit 113-10 Ground
 - 117 Data bit 114-10 Ground
 - 118 Data bit 115-10 Ground
 - 119 Data bit 116-10 Ground
 - 120 Data bit 117-10 Ground
 - 121 Data bit 118-10 Ground
 - 122 Data bit 119-10 Ground
 - 123 Data bit 120-10 Ground
 - 124 Data bit 121-10 Ground
 - 125 Data bit 122-10 Ground
 - 126 Data bit 123-10 Ground
 - 127 Data bit 124-10 Ground
 - 128 Data bit 125-10 Ground
 - 129 Data bit 126-10 Ground
 - 130 Data bit 127-10 Ground
 - 131 Data bit 128-10 Ground
 - 132 Data bit 129-10 Ground
 - 133 Data bit 130-10 Ground
 - 134 Data bit 131-10 Ground
 - 135 Data bit 132-10 Ground
 - 136 Data bit 133-10 Ground
 - 137 Data bit 134-10 Ground
 - 138 Data bit 135-10 Ground
 - 139 Data bit 136-10 Ground
 - 140 Data bit 137-10 Ground
 - 141 Data bit 138-10 Ground
 - 142 Data bit 139-10 Ground
 - 143 Data bit 140-10 Ground
 - 144 Data bit 141-10 Ground
 - 145 Data bit 142-10 Ground
 - 146 Data bit 143-10 Ground
 - 147 Data bit 144-10 Ground
 - 148 Data bit 145-10 Ground
 - 149 Data bit 146-10 Ground
 - 150 Data bit 147-10 Ground
 - 151 Data bit 148-10 Ground
 - 152 Data bit 149-10 Ground
 - 153 Data bit 150-10 Ground
 - 154 Data bit 151-10 Ground
 - 155 Data bit 152-10 Ground
 - 156 Data bit 153-10 Ground
 - 157 Data bit 154-10 Ground
 - 158 Data bit 155-10 Ground
 - 159 Data bit 156-10 Ground
 - 160 Data bit 157-10 Ground
 - 161 Data bit 158-10 Ground
 - 162 Data bit 159-10 Ground
 - 163 Data bit 160-10 Ground
 - 164 Data bit 161-10 Ground
 - 165 Data bit 162-10 Ground
 - 166 Data bit 163-10 Ground
 - 167 Data bit 164-10 Ground
 - 168 Data bit 165-10 Ground
 - 169 Data bit 166-10 Ground
 - 170 Data bit 167-10 Ground
 - 171 Data bit 168-10 Ground
 - 172 Data bit 169-10 Ground
 - 173 Data bit 170-10 Ground
 - 174 Data bit 171-10 Ground
 - 175 Data bit 172-10 Ground
 - 176 Data bit 173-10 Ground
 - 177 Data bit 174-10 Ground
 - 178 Data bit 175-10 Ground
 - 179 Data bit 176-10 Ground
 - 180 Data bit 177-10 Ground
 - 181 Data bit 178-10 Ground
 - 182 Data bit 179-10 Ground
 - 183 Data bit 180-10 Ground
 - 184 Data bit 181-10 Ground
 - 185 Data bit 182-10 Ground
 - 186 Data bit 183-10 Ground
 - 187 Data bit 184-10 Ground
 - 188 Data bit 185-10 Ground
 - 189 Data bit 186-10 Ground
 - 190 Data bit 187-10 Ground
 - 191 Data bit 188-10 Ground
 - 192 Data bit 189-10 Ground
 - 193 Data bit 190-10 Ground
 - 194 Data bit 191-10 Ground
 - 195 Data bit 192-10 Ground
 - 196 Data bit 193-10 Ground
 - 197 Data bit 194-10 Ground
 - 198 Data bit 195-10 Ground
 - 199 Data bit 196-10 Ground
 - 200 Data bit 197-10 Ground
 - 201 Data bit 198-10 Ground
 - 202 Data bit 199-10 Ground
 - 203 Data bit 200-10 Ground
 - 204 Data bit 201-10 Ground
 - 205 Data bit 202-10 Ground
 - 206 Data bit 203-10 Ground
 - 207 Data bit 204-10 Ground
 - 208 Data bit 205-10 Ground
 - 209 Data bit 206-10 Ground
 - 210 Data bit 207-10 Ground
 - 211 Data bit 208-10 Ground
 - 212 Data bit 209-10 Ground
 - 213 Data bit 210-10 Ground
 - 214 Data bit 211-10 Ground
 - 215 Data bit 212-10 Ground
 - 216 Data bit 213-10 Ground
 - 217 Data bit 214-10 Ground
 - 218 Data bit 215-10 Ground
 - 219 Data bit 216-10 Ground
 - 220 Data bit 217-10 Ground
 - 221 Data bit 218-10 Ground
 - 222 Data bit 219-10 Ground
 - 223 Data bit 220-10 Ground
 - 224 Data bit 221-10 Ground
 - 225 Data bit 222-10 Ground
 - 226 Data bit 223-10 Ground
 - 227 Data bit 224-10 Ground
 - 228 Data bit 225-10 Ground
 - 229 Data bit 226-10 Ground
 - 230 Data bit 227-10 Ground
 - 231 Data bit 228-10 Ground
 - 232 Data bit 229-10 Ground
 - 233 Data bit 230-10 Ground
 - 234 Data bit 231-10 Ground
 - 235 Data bit 232-10 Ground
 - 236 Data bit 233-10 Ground
 - 237 Data bit 234-10 Ground
 - 238 Data bit 235-10 Ground
 - 239 Data bit 236-10 Ground
 - 240 Data bit 237-10 Ground
 - 241 Data bit 238-10 Ground
 - 242 Data bit 239-10 Ground
 - 243 Data bit 240-10 Ground
 - 244 Data bit 241-10 Ground
 - 245 Data bit 242-10 Ground
 - 246 Data bit 243-10 Ground
 - 247 Data bit 244-10 Ground
 - 248 Data bit 245-10 Ground
 - 249 Data bit 246-10 Ground
 - 250 Data bit 247-10 Ground
 - 251 Data bit 248-10 Ground
 - 252 Data bit 249-10 Ground
 - 253 Data bit 250-10 Ground
 - 254 Data bit 251-10 Ground
 - 255 Data bit 252-10 Ground
 - 256 Data bit 253-10 Ground
 - 257 Data bit 254-10 Ground
 - 258 Data bit 255-10 Ground
 - 259 Data bit 256-10 Ground
 - 260 Data bit 257-10 Ground
 - 261 Data bit 258-10 Ground
 - 262 Data bit 259-10 Ground
 - 263 Data bit 260-10 Ground
 - 264 Data bit 261-10 Ground
 - 265 Data bit 262-10 Ground
 - 266 Data bit 263-10 Ground
 - 267 Data bit 264-10 Ground
 - 268 Data bit 265-10 Ground
 - 269 Data bit 266-10 Ground
 - 270 Data bit 267-10 Ground
 - 271 Data bit 268-10 Ground
 - 272 Data bit 269-10 Ground
 - 273 Data bit 270-10 Ground
 - 274 Data bit 271-10 Ground
 - 275 Data bit 272-10 Ground
 - 276 Data bit 273-10 Ground
 - 277 Data bit 274-10 Ground
 - 278 Data bit 275-10 Ground
 - 279 Data bit 276-10 Ground
 - 280 Data bit 277-10 Ground
 - 281 Data bit 278-10 Ground
 - 282 Data bit 279-10 Ground
 - 283 Data bit 280-10 Ground
 - 284 Data bit 281-10 Ground
 - 285 Data bit 282-10 Ground
 - 286 Data bit 283-10 Ground
 - 287 Data bit 284-10 Ground
 - 288 Data bit 285-10 Ground
 - 289 Data bit 286-10 Ground
 - 290 Data bit 287-10 Ground
 - 291 Data bit 288-10 Ground
 - 292 Data bit 289-10 Ground
 - 293 Data bit 290-10 Ground
 - 294 Data bit 291-10 Ground
 - 295 Data bit 292-10 Ground
 - 296 Data bit 293-10 Ground
 - 297 Data bit 294-10 Ground
 - 298 Data bit 295-10 Ground
 - 299 Data bit 296-10 Ground
 - 300 Data bit 297-10 Ground
 - 301 Data bit 298-10 Ground
 - 302 Data bit 299-10 Ground
 - 303 Data bit 300-10 Ground
 - 304 Data bit 301-10 Ground
 - 305 Data bit 302-10 Ground
 - 306 Data bit 303-10 Ground
 - 307 Data bit 304-10 Ground
 - 308 Data bit 305-10 Ground
 - 309 Data bit 306-10 Ground
 - 310 Data bit 307-10 Ground
 - 311 Data bit 308-10 Ground
 - 312 Data bit 309-10 Ground
 - 313 Data bit 310-10 Ground
 - 314 Data bit 311-10 Ground
 - 315 Data bit 312-10 Ground
 - 316 Data bit 313-10 Ground
 - 317 Data bit 314-10 Ground
 - 318 Data bit 315-10 Ground
 - 319 Data bit 316-10 Ground
 - 320 Data bit 317-10 Ground
 - 321 Data bit 318-10 Ground
 - 322 Data bit 319-10 Ground
 - 323 Data bit 320-10 Ground
 - 324 Data bit 321-10 Ground
 - 325 Data bit 322-10 Ground
 - 326 Data bit 323-10 Ground
 - 327 Data bit 324-10 Ground
 - 328 Data bit 325-10 Ground
 - 329 Data bit 326-10 Ground
 - 330 Data bit 327-10 Ground
 - 331 Data bit 328-10 Ground
 - 332 Data bit 329-10 Ground
 - 333 Data bit 330-10 Ground
 - 334 Data bit 331-10 Ground
 - 335 Data bit 332-10 Ground
 - 336 Data bit 333-10 Ground
 - 337 Data bit 334-10 Ground
 - 338 Data bit 335-10 Ground
 - 339 Data bit 336-10 Ground
 - 340 Data bit 337-10 Ground
 - 341 Data bit 338-10 Ground
 - 342 Data bit 339-10 Ground
 - 343 Data bit 340-10 Ground
 - 344 Data bit 341-10 Ground
 - 345 Data bit 342-10 Ground
 - 346 Data bit 343-10 Ground
 - 347 Data bit 344-10 Ground
 - 348 Data bit 345-10 Ground
 - 349 Data bit 346-10 Ground
 - 350 Data bit 347-10 Ground
 - 351 Data bit 348-10 Ground
 - 352 Data bit 349-10 Ground
 - 353 Data bit 350-10 Ground
 - 354 Data bit 351-10 Ground
 - 355 Data bit 352-10 Ground
 - 356 Data bit 353-10 Ground
 - 357 Data bit 354-10 Ground
 - 358 Data bit 355-10 Ground
 - 359 Data bit 356-10 Ground
 - 360 Data bit 357-10 Ground
 - 361 Data bit 358-10 Ground
 - 362 Data bit 359-10 Ground
 - 363 Data bit 360-10 Ground
 - 364 Data bit 361-10 Ground
 - 365 Data bit 362-10 Ground
 - 366 Data bit 363-10 Ground
 - 367 Data bit 364-10 Ground
 - 368 Data bit 365-10 Ground
 - 369 Data bit 366-10 Ground
 - 370 Data bit 367-10 Ground
 - 371 Data bit 368-10 Ground
 - 372 Data bit 369-10 Ground
 - 373 Data bit 370-10 Ground
 - 374 Data bit 371-10 Ground
 - 375 Data bit 372-10 Ground
 - 376 Data bit 373-10 Ground
 - 377 Data bit 374-10 Ground
 - 378 Data bit 375-10 Ground
 - 379 Data bit 376-10 Ground
 - 380 Data bit 377-10 Ground
 - 381 Data bit 378-10 Ground
 - 382 Data bit 379-10 Ground
 - 383 Data bit 380-10 Ground
 - 384 Data bit 381-10 Ground
 - 385 Data bit 382-10 Ground
 - 386 Data bit 383-10 Ground
 - 387 Data bit 384-10 Ground
 - 388 Data bit 385-10 Ground
 - 389 Data bit 386-10 Ground
 - 390 Data bit 387-10 Ground
 - 391 Data bit 388-10 Ground
 - 392 Data bit 389-10 Ground
 - 393 Data bit 390-10 Ground
 - 394 Data bit 391-10 Ground
 - 395 Data bit 392-10 Ground
 - 396 Data bit 393-10 Ground
 - 397 Data bit 394-10 Ground
 - 398 Data bit 395-10 Ground
 - 399 Data bit 396-10 Ground
 - 400 Data bit 397-10 Ground
 - 401 Data bit 398-10 Ground
 - 402 Data bit 399-10 Ground
 - 403 Data bit 400-10 Ground
 - 404 Data bit 401-10 Ground
 - 405 Data bit 402-10 Ground
 - 406 Data bit 403-10 Ground
 - 407 Data bit 404-10 Ground
 - 408 Data bit 405-10 Ground
 - 409 Data bit 406-10 Ground
 - 410 Data bit 407-10 Ground
 - 411 Data bit 408-10 Ground
 - 412 Data bit 409-10 Ground
 - 413 Data bit 410-10 Ground
 - 414 Data bit 411-10 Ground
 - 415 Data bit 412-10 Ground
 - 416 Data bit 413-10 Ground
 - 417 Data bit 414-10 Ground
 - 418 Data bit 415-10 Ground
 - 419 Data bit 416-10 Ground
 - 420 Data bit 417-10 Ground
 - 421 Data bit 418-10 Ground
 - 422 Data bit 419-10 Ground
 - 423 Data bit 420-10 Ground
 - 424 Data bit 421-10 Ground
 - 425 Data bit 422-10 Ground
 - 426 Data bit 423-10 Ground
 - 427 Data bit 424-10 Ground
 - 428 Data bit 425-10 Ground
 - 429 Data bit 426-10 Ground
 - 430 Data bit 427-10 Ground
 - 431 Data bit 428-10 Ground
 - 432 Data bit 429-10 Ground
 - 433 Data bit 430-10 Ground
 - 434 Data bit 431-10 Ground
 - 435 Data bit 432-10 Ground
 - 436 Data bit 433-10 Ground
 - 437 Data bit 434-10 Ground
 - 438 Data bit 435-10 Ground
 - 439 Data bit 436-10 Ground
 - 440 Data bit 437-10 Ground
 - 441 Data bit 438-10 Ground
 - 442 Data bit 439-10 Ground
 - 443 Data bit 440-10 Ground
 - 444 Data bit 441-10 Ground
 - 445 Data bit 442-10 Ground
 - 446 Data bit 443-10 Ground
 - 447 Data bit 444-10 Ground
 - 448 Data bit 445-10 Ground
 - 449 Data bit 446-10 Ground
 - 450 Data bit 447-10 Ground
 - 451 Data bit 448-10 Ground
 - 452 Data bit 449-10 Ground
 - 453 Data bit 450-10 Ground
 - 454 Data bit 451-10 Ground
 - 455 Data bit 452-10 Ground
 - 456 Data bit 453-10 Ground
 - 457 Data bit 454-10 Ground
 - 458 Data bit 455-10 Ground
 - 459 Data bit 456-10 Ground
 - 460 Data bit 457-10 Ground
 - 461 Data bit 458-10 Ground
 - 462 Data bit 459-10 Ground
 - 463 Data bit 460-10 Ground
 - 464 Data bit 461-10 Ground
 - 465 Data bit 462-10 Ground
 - 466 Data bit 463-10 Ground
 - 467 Data bit 464-10 Ground
 - 468 Data bit 465-10 Ground
 - 469 Data bit 466-10 Ground
 - 470 Data bit 467-10 Ground
 - 471 Data bit 468-10 Ground
 - 472 Data bit 469-10 Ground
 - 473 Data bit 470-10 Ground
 - 474 Data bit 471-10 Ground
 - 475 Data bit 472-10 Ground
 - 476 Data bit 473-10 Ground
 - 477 Data bit 474-10 Ground
 - 478 Data bit 475-10 Ground
 - 479 Data bit 476-10 Ground
 - 480 Data bit 477-10 Ground
 - 481 Data bit 478-10 Ground
 - 482 Data bit 479-10 Ground
 - 483 Data bit 480-10 Ground
 - 484 Data bit 481-10 Ground
 - 485 Data bit 482-10 Ground
 - 486 Data bit 483-10 Ground
 - 487 Data bit 484-10 Ground
 - 488 Data bit 485-10 Ground
 - 489 Data bit 486-10 Ground
 - 490 Data bit 487-10 Ground
 - 491 Data bit 488-10 Ground
 - 492 Data bit 489-10 Ground
 - 493 Data bit 490-10 Ground
 - 494 Data bit 491-10 Ground
 - 495 Data bit 492-10 Ground
 - 496 Data bit 493-10 Ground
 - 497 Data bit 494-10 Ground
 - 498 Data bit 495-10 Ground
 - 499 Data bit 496-10 Ground
 - 500 Data bit 497-10 Ground
 - 501 Data bit 498-10 Ground
 - 502 Data bit 499-10 Ground
 - 503 Data bit 500-10 Ground
 - 504 Data bit 501-10 Ground
 - 505 Data bit 502-10 Ground
 - 506 Data bit 503-10 Ground
 - 507 Data bit 504-10 Ground
 - 508 Data bit 505-10 Ground
 - 509 Data bit 506-10 Ground
 - 510 Data bit 507-10 Ground
 - 511 Data bit 508-10 Ground
 - 512 Data bit 509-10 Ground
 - 513 Data bit 510-10 Ground
 - 514 Data bit 511-10 Ground
 - 515 Data bit 512-10 Ground
 - 516 Data bit 513-10 Ground
 - 517 Data bit 514-10 Ground
 - 518 Data bit 515-10 Ground
 - 519 Data bit 516-10 Ground
 - 520 Data bit 517-10 Ground

Winners and Losers

Every month, Gordon Lee will look at some prize programming points from a previous month's competition.

OK, you lucky lot! Here is your opportunity to share in some feedback from Dragon User's regular competition pages. For an experimental period, the Powers that Be (We call them the Powers, That Be, ancient now — G.) have asked me to sift through the entries to the competition answers and highlight some of the innovations in programming and problem solving that competitors use. These should provide a useful reference for use in programming generally, as well as for anticipating future competitions (I can see that I shall have to be even more decisive...). In addition, mention of some of the commonly occurring mistakes should be of help in avoiding these pitfalls, especially for beginners who may be susceptible to having 'go' at the competitions, and maybe win some prizes.

This month, we'll be looking at the August '86 competition in greater detail, and, as with most of these problems, there are four main stages in their solution:

1 Read the question carefully. How many times have we heard that bit of advice? For the August problem you had to substitute digits for the letters in DRAGON USER, such that the result was a two-digit number, the value of which spelled a common English word when the digits were replaced by letters from the original sum.

A surprisingly large number of competitors were of the impression that you were not permitted to use a zero as one of the digits. There was nothing in the question to suggest this, although in the event, this would not have affected the result which, purely by chance, did not contain a zero. Had it done so, many of the programs submitted would not have come up with the answer. The question of a 'leading zero' is a different matter. Conventionally, in puzzles of this type, it is assumed that a leading zero is not permitted, unless specifically mentioned in the question. Thus, in this particular question, neither the 'D' nor the 'U' can be replaced by a zero, since it would seem reasonable to assume that, for example, 'DRAGON' was a six-digit number and not a five-digit number with a zero in front.

A number of entrants allowed values with a leading zero which, apart from increasing the number of tests to be carried out, could have produced spurious results.

2 Method of approach, or the interpretation of the problem into a program which the computer can handle. Just as there is more than one way to cook a cat so there may well be alternative ways, often as good, maybe better, to solve a given problem. Heavily all competitors entered, probably wisely, to take as a starting point the possible values of USER and the two-digit quotient, and then work backwards to find and test the corresponding value for 'DRAGON'. Most entrants using this approach generated the value for 'USER' in a single FOR/NEXT loop, testing each value for duplication of digits, but some used four FOR/NEXT loops to generate and test each digit individually before combining them into the final four-digit number. Which of the two methods will be most time-efficient is best determined by experiment. The long running time of the programs was a generally expressed comment by many readers, so any opportunity to reduce this should be seized. However, it is probably more efficient to exclude whole ranges of impossible values for testing rather than to test every set of values for comparatively minor features. The testing on this page is a 'composite' compiled from a number of listings submitted, but which follows the general ideas of the majority of the programs submitted. It runs for several hours, but the winning answer appears after about three Cee-Enn hours from earlier competition set in Dragon User admitted to a running period of over a week! Clearly it's a case of procrastination and, being of a lazy nature myself, I see no reason why a computer should not do most of the work — that's what they are there for, isn't it?

Most entrants took the minimum and maximum values of 'USER' to be 1000 and 9999 respectively. In fact, 'USER' must be greater than 2910 as any smaller value, multiplied by an two-digit number will give as a value for 'DRAGON' either a five-digit

number or, at best, a six-digit number beginning with a '1'. Clearly, if the '1' already stands for the 'U' then it cannot also represent the 'D'. The range to be included can be further reduced in many cases, if 'U' is the two-digit result of DRAGON/USER, then any value of 'U' in the range of USER will be between INT(1000/4916) and INT(9999/4916). Other points which can be incorporated into the program are that 'U' cannot be a multiple of ten or its cube will end in three zeros, and hence the resulting word in three identical letters. Also, 'U' cannot equal zero, otherwise 'U' would also have to equal zero if the quotient is to be a whole number. From this analysis of the problem we should now be in a position to:

3 Write the program. The listing given here will print out all sets of values in which the cube without has all its digits present in the generating division, the latter equivalents being printed alongside. A few (very few) readers also tried to quicken things along by using the notorious Dragon speed piece, Am FPG98. The effectiveness of the individual programming submitted was reflected in the number of possible 'words' that were printed out for examination. Where included, these varied from a list of under a dozen words to several with over seventy. Also the length of the programs ranged from under 10 to over 80 program lines. Once these lists have been printed out, the final (and comparatively easy) stage is to:

4 Check the results. In the case of the competition it question this was simply to find a common English word from amongst these letters sets printed out. R. A. Newman at Banbury reports that the letter set GPRMNN occurs twice (has this word any special or mystical significance, EAT?), while C. Hinchman of Middleborough also lists a number of alternative solutions. These include such words as samms, saucos, sweats and rem, which he describes as 'words used in comics to give sound effects in fights being crushed, people being tortured in mercilessly devious, and airtightening shot down'. Strange reading matter you have there! Phil Sapio of Liverpool was surprised to find that there was only one acceptable word formed from the cubes tested. In setting the question, I had first tested the squares without finding a single word. As this type of letter substitution is almost the equivalent of a pseudo-random selection of letters from the words 'DRAGON USER' the chance of an English word occurring at all must be fairly low. Perhaps interested readers would like to generate some random sequences and find the incidence of acceptable words.

Finally, Martin Wingo of Banbury deserves mention for the month's worst double pun used in the tie-breaker sentence:

"You're never alone in the Universe... Because Mars is always got Chris' plod" Go on! Say it quickly! Now I know what made Banbury cross!

```
100 DIM J(1000000)
101 FOR I=1 TO 99
102 IF A(UBOUND(I))=0 THEN FOR J=1 TO 99
103 IF A(UBOUND(J))=0 THEN FOR K=1 TO 99
104 IF A(UBOUND(K))=0 THEN FOR L=1 TO 99
105 IF A(UBOUND(L))=0 THEN FOR M=1 TO 99
106 IF A(UBOUND(M))=0 THEN FOR N=1 TO 99
107 IF A(UBOUND(N))=0 THEN FOR O=1 TO 99
108 IF A(UBOUND(O))=0 THEN FOR P=1 TO 99
109 IF A(UBOUND(P))=0 THEN FOR Q=1 TO 99
110 IF A(UBOUND(Q))=0 THEN FOR R=1 TO 99
111 IF A(UBOUND(R))=0 THEN FOR S=1 TO 99
112 IF A(UBOUND(S))=0 THEN FOR T=1 TO 99
113 IF A(UBOUND(T))=0 THEN FOR U=1 TO 99
114 IF A(UBOUND(U))=0 THEN FOR V=1 TO 99
115 IF A(UBOUND(V))=0 THEN FOR W=1 TO 99
116 IF A(UBOUND(W))=0 THEN FOR X=1 TO 99
117 IF A(UBOUND(X))=0 THEN FOR Y=1 TO 99
118 IF A(UBOUND(Y))=0 THEN FOR Z=1 TO 99
119 IF A(UBOUND(Z))=0 THEN FOR A=1 TO 99
120 IF A(UBOUND(A))=0 THEN FOR B=1 TO 99
121 IF A(UBOUND(B))=0 THEN FOR C=1 TO 99
122 IF A(UBOUND(C))=0 THEN FOR D=1 TO 99
123 IF A(UBOUND(D))=0 THEN FOR E=1 TO 99
124 IF A(UBOUND(E))=0 THEN FOR F=1 TO 99
125 IF A(UBOUND(F))=0 THEN FOR G=1 TO 99
126 IF A(UBOUND(G))=0 THEN FOR H=1 TO 99
127 IF A(UBOUND(H))=0 THEN FOR I=1 TO 99
128 IF A(UBOUND(I))=0 THEN FOR J=1 TO 99
129 IF A(UBOUND(J))=0 THEN FOR K=1 TO 99
130 IF A(UBOUND(K))=0 THEN FOR L=1 TO 99
131 IF A(UBOUND(L))=0 THEN FOR M=1 TO 99
132 IF A(UBOUND(M))=0 THEN FOR N=1 TO 99
133 IF A(UBOUND(N))=0 THEN FOR O=1 TO 99
134 IF A(UBOUND(O))=0 THEN FOR P=1 TO 99
135 IF A(UBOUND(P))=0 THEN FOR Q=1 TO 99
136 IF A(UBOUND(Q))=0 THEN FOR R=1 TO 99
137 IF A(UBOUND(R))=0 THEN FOR S=1 TO 99
138 IF A(UBOUND(S))=0 THEN FOR T=1 TO 99
139 IF A(UBOUND(T))=0 THEN FOR U=1 TO 99
140 IF A(UBOUND(U))=0 THEN FOR V=1 TO 99
141 IF A(UBOUND(V))=0 THEN FOR W=1 TO 99
142 IF A(UBOUND(W))=0 THEN FOR X=1 TO 99
143 IF A(UBOUND(X))=0 THEN FOR Y=1 TO 99
144 IF A(UBOUND(Y))=0 THEN FOR Z=1 TO 99
145 IF A(UBOUND(Z))=0 THEN FOR A=1 TO 99
146 IF A(UBOUND(A))=0 THEN FOR B=1 TO 99
147 IF A(UBOUND(B))=0 THEN FOR C=1 TO 99
148 IF A(UBOUND(C))=0 THEN FOR D=1 TO 99
149 IF A(UBOUND(D))=0 THEN FOR E=1 TO 99
150 IF A(UBOUND(E))=0 THEN FOR F=1 TO 99
151 IF A(UBOUND(F))=0 THEN FOR G=1 TO 99
152 IF A(UBOUND(G))=0 THEN FOR H=1 TO 99
153 IF A(UBOUND(H))=0 THEN FOR I=1 TO 99
154 IF A(UBOUND(I))=0 THEN FOR J=1 TO 99
155 IF A(UBOUND(J))=0 THEN FOR K=1 TO 99
156 IF A(UBOUND(K))=0 THEN FOR L=1 TO 99
157 IF A(UBOUND(L))=0 THEN FOR M=1 TO 99
158 IF A(UBOUND(M))=0 THEN FOR N=1 TO 99
159 IF A(UBOUND(N))=0 THEN FOR O=1 TO 99
160 IF A(UBOUND(O))=0 THEN FOR P=1 TO 99
161 IF A(UBOUND(P))=0 THEN FOR Q=1 TO 99
162 IF A(UBOUND(Q))=0 THEN FOR R=1 TO 99
163 IF A(UBOUND(R))=0 THEN FOR S=1 TO 99
164 IF A(UBOUND(S))=0 THEN FOR T=1 TO 99
165 IF A(UBOUND(T))=0 THEN FOR U=1 TO 99
166 IF A(UBOUND(U))=0 THEN FOR V=1 TO 99
167 IF A(UBOUND(V))=0 THEN FOR W=1 TO 99
168 IF A(UBOUND(W))=0 THEN FOR X=1 TO 99
169 IF A(UBOUND(X))=0 THEN FOR Y=1 TO 99
170 IF A(UBOUND(Y))=0 THEN FOR Z=1 TO 99
171 IF A(UBOUND(Z))=0 THEN FOR A=1 TO 99
172 IF A(UBOUND(A))=0 THEN FOR B=1 TO 99
173 IF A(UBOUND(B))=0 THEN FOR C=1 TO 99
174 IF A(UBOUND(C))=0 THEN FOR D=1 TO 99
175 IF A(UBOUND(D))=0 THEN FOR E=1 TO 99
176 IF A(UBOUND(E))=0 THEN FOR F=1 TO 99
177 IF A(UBOUND(F))=0 THEN FOR G=1 TO 99
178 IF A(UBOUND(G))=0 THEN FOR H=1 TO 99
179 IF A(UBOUND(H))=0 THEN FOR I=1 TO 99
180 IF A(UBOUND(I))=0 THEN FOR J=1 TO 99
181 IF A(UBOUND(J))=0 THEN FOR K=1 TO 99
182 IF A(UBOUND(K))=0 THEN FOR L=1 TO 99
183 IF A(UBOUND(L))=0 THEN FOR M=1 TO 99
184 IF A(UBOUND(M))=0 THEN FOR N=1 TO 99
185 IF A(UBOUND(N))=0 THEN FOR O=1 TO 99
186 IF A(UBOUND(O))=0 THEN FOR P=1 TO 99
187 IF A(UBOUND(P))=0 THEN FOR Q=1 TO 99
188 IF A(UBOUND(Q))=0 THEN FOR R=1 TO 99
189 IF A(UBOUND(R))=0 THEN FOR S=1 TO 99
190 IF A(UBOUND(S))=0 THEN FOR T=1 TO 99
191 IF A(UBOUND(T))=0 THEN FOR U=1 TO 99
192 IF A(UBOUND(U))=0 THEN FOR V=1 TO 99
193 IF A(UBOUND(V))=0 THEN FOR W=1 TO 99
194 IF A(UBOUND(W))=0 THEN FOR X=1 TO 99
195 IF A(UBOUND(X))=0 THEN FOR Y=1 TO 99
196 IF A(UBOUND(Y))=0 THEN FOR Z=1 TO 99
197 IF A(UBOUND(Z))=0 THEN FOR A=1 TO 99
198 IF A(UBOUND(A))=0 THEN FOR B=1 TO 99
199 IF A(UBOUND(B))=0 THEN FOR C=1 TO 99
200 IF A(UBOUND(C))=0 THEN FOR D=1 TO 99
201 IF A(UBOUND(D))=0 THEN FOR E=1 TO 99
202 IF A(UBOUND(E))=0 THEN FOR F=1 TO 99
203 IF A(UBOUND(F))=0 THEN FOR G=1 TO 99
204 IF A(UBOUND(G))=0 THEN FOR H=1 TO 99
205 IF A(UBOUND(H))=0 THEN FOR I=1 TO 99
206 IF A(UBOUND(I))=0 THEN FOR J=1 TO 99
207 IF A(UBOUND(J))=0 THEN FOR K=1 TO 99
208 IF A(UBOUND(K))=0 THEN FOR L=1 TO 99
209 IF A(UBOUND(L))=0 THEN FOR M=1 TO 99
210 IF A(UBOUND(M))=0 THEN FOR N=1 TO 99
211 IF A(UBOUND(N))=0 THEN FOR O=1 TO 99
212 IF A(UBOUND(O))=0 THEN FOR P=1 TO 99
213 IF A(UBOUND(P))=0 THEN FOR Q=1 TO 99
214 IF A(UBOUND(Q))=0 THEN FOR R=1 TO 99
215 IF A(UBOUND(R))=0 THEN FOR S=1 TO 99
216 IF A(UBOUND(S))=0 THEN FOR T=1 TO 99
217 IF A(UBOUND(T))=0 THEN FOR U=1 TO 99
218 IF A(UBOUND(U))=0 THEN FOR V=1 TO 99
219 IF A(UBOUND(V))=0 THEN FOR W=1 TO 99
220 IF A(UBOUND(W))=0 THEN FOR X=1 TO 99
221 IF A(UBOUND(X))=0 THEN FOR Y=1 TO 99
222 IF A(UBOUND(Y))=0 THEN FOR Z=1 TO 99
223 IF A(UBOUND(Z))=0 THEN FOR A=1 TO 99
224 IF A(UBOUND(A))=0 THEN FOR B=1 TO 99
225 IF A(UBOUND(B))=0 THEN FOR C=1 TO 99
226 IF A(UBOUND(C))=0 THEN FOR D=1 TO 99
227 IF A(UBOUND(D))=0 THEN FOR E=1 TO 99
228 IF A(UBOUND(E))=0 THEN FOR F=1 TO 99
229 IF A(UBOUND(F))=0 THEN FOR G=1 TO 99
230 IF A(UBOUND(G))=0 THEN FOR H=1 TO 99
231 IF A(UBOUND(H))=0 THEN FOR I=1 TO 99
232 IF A(UBOUND(I))=0 THEN FOR J=1 TO 99
233 IF A(UBOUND(J))=0 THEN FOR K=1 TO 99
234 IF A(UBOUND(K))=0 THEN FOR L=1 TO 99
235 IF A(UBOUND(L))=0 THEN FOR M=1 TO 99
236 IF A(UBOUND(M))=0 THEN FOR N=1 TO 99
237 IF A(UBOUND(N))=0 THEN FOR O=1 TO 99
238 IF A(UBOUND(O))=0 THEN FOR P=1 TO 99
239 IF A(UBOUND(P))=0 THEN FOR Q=1 TO 99
240 IF A(UBOUND(Q))=0 THEN FOR R=1 TO 99
241 IF A(UBOUND(R))=0 THEN FOR S=1 TO 99
242 IF A(UBOUND(S))=0 THEN FOR T=1 TO 99
243 IF A(UBOUND(T))=0 THEN FOR U=1 TO 99
244 IF A(UBOUND(U))=0 THEN FOR V=1 TO 99
245 IF A(UBOUND(V))=0 THEN FOR W=1 TO 99
246 IF A(UBOUND(W))=0 THEN FOR X=1 TO 99
247 IF A(UBOUND(X))=0 THEN FOR Y=1 TO 99
248 IF A(UBOUND(Y))=0 THEN FOR Z=1 TO 99
249 IF A(UBOUND(Z))=0 THEN FOR A=1 TO 99
250 IF A(UBOUND(A))=0 THEN FOR B=1 TO 99
251 IF A(UBOUND(B))=0 THEN FOR C=1 TO 99
252 IF A(UBOUND(C))=0 THEN FOR D=1 TO 99
253 IF A(UBOUND(D))=0 THEN FOR E=1 TO 99
254 IF A(UBOUND(E))=0 THEN FOR F=1 TO 99
255 IF A(UBOUND(F))=0 THEN FOR G=1 TO 99
256 IF A(UBOUND(G))=0 THEN FOR H=1 TO 99
257 IF A(UBOUND(H))=0 THEN FOR I=1 TO 99
258 IF A(UBOUND(I))=0 THEN FOR J=1 TO 99
259 IF A(UBOUND(J))=0 THEN FOR K=1 TO 99
260 IF A(UBOUND(K))=0 THEN FOR L=1 TO 99
261 IF A(UBOUND(L))=0 THEN FOR M=1 TO 99
262 IF A(UBOUND(M))=0 THEN FOR N=1 TO 99
263 IF A(UBOUND(N))=0 THEN FOR O=1 TO 99
264 IF A(UBOUND(O))=0 THEN FOR P=1 TO 99
265 IF A(UBOUND(P))=0 THEN FOR Q=1 TO 99
266 IF A(UBOUND(Q))=0 THEN FOR R=1 TO 99
267 IF A(UBOUND(R))=0 THEN FOR S=1 TO 99
268 IF A(UBOUND(S))=0 THEN FOR T=1 TO 99
269 IF A(UBOUND(T))=0 THEN FOR U=1 TO 99
270 IF A(UBOUND(U))=0 THEN FOR V=1 TO 99
271 IF A(UBOUND(V))=0 THEN FOR W=1 TO 99
272 IF A(UBOUND(W))=0 THEN FOR X=1 TO 99
273 IF A(UBOUND(X))=0 THEN FOR Y=1 TO 99
274 IF A(UBOUND(Y))=0 THEN FOR Z=1 TO 99
275 IF A(UBOUND(Z))=0 THEN FOR A=1 TO 99
276 IF A(UBOUND(A))=0 THEN FOR B=1 TO 99
277 IF A(UBOUND(B))=0 THEN FOR C=1 TO 99
278 IF A(UBOUND(C))=0 THEN FOR D=1 TO 99
279 IF A(UBOUND(D))=0 THEN FOR E=1 TO 99
280 IF A(UBOUND(E))=0 THEN FOR F=1 TO 99
281 IF A(UBOUND(F))=0 THEN FOR G=1 TO 99
282 IF A(UBOUND(G))=0 THEN FOR H=1 TO 99
283 IF A(UBOUND(H))=0 THEN FOR I=1 TO 99
284 IF A(UBOUND(I))=0 THEN FOR J=1 TO 99
285 IF A(UBOUND(J))=0 THEN FOR K=1 TO 99
286 IF A(UBOUND(K))=0 THEN FOR L=1 TO 99
287 IF A(UBOUND(L))=0 THEN FOR M=1 TO 99
288 IF A(UBOUND(M))=0 THEN FOR N=1 TO 99
289 IF A(UBOUND(N))=0 THEN FOR O=1 TO 99
290 IF A(UBOUND(O))=0 THEN FOR P=1 TO 99
291 IF A(UBOUND(P))=0 THEN FOR Q=1 TO 99
292 IF A(UBOUND(Q))=0 THEN FOR R=1 TO 99
293 IF A(UBOUND(R))=0 THEN FOR S=1 TO 99
294 IF A(UBOUND(S))=0 THEN FOR T=1 TO 99
295 IF A(UBOUND(T))=0 THEN FOR U=1 TO 99
296 IF A(UBOUND(U))=0 THEN FOR V=1 TO 99
297 IF A(UBOUND(V))=0 THEN FOR W=1 TO 99
298 IF A(UBOUND(W))=0 THEN FOR X=1 TO 99
299 IF A(UBOUND(X))=0 THEN FOR Y=1 TO 99
300 IF A(UBOUND(Y))=0 THEN FOR Z=1 TO 99
301 IF A(UBOUND(Z))=0 THEN FOR A=1 TO 99
302 IF A(UBOUND(A))=0 THEN FOR B=1 TO 99
303 IF A(UBOUND(B))=0 THEN FOR C=1 TO 99
304 IF A(UBOUND(C))=0 THEN FOR D=1 TO 99
305 IF A(UBOUND(D))=0 THEN FOR E=1 TO 99
306 IF A(UBOUND(E))=0 THEN FOR F=1 TO 99
307 IF A(UBOUND(F))=0 THEN FOR G=1 TO 99
308 IF A(UBOUND(G))=0 THEN FOR H=1 TO 99
309 IF A(UBOUND(H))=0 THEN FOR I=1 TO 99
310 IF A(UBOUND(I))=0 THEN FOR J=1 TO 99
311 IF A(UBOUND(J))=0 THEN FOR K=1 TO 99
312 IF A(UBOUND(K))=0 THEN FOR L=1 TO 99
313 IF A(UBOUND(L))=0 THEN FOR M=1 TO 99
314 IF A(UBOUND(M))=0 THEN FOR N=1 TO 99
315 IF A(UBOUND(N))=0 THEN FOR O=1 TO 99
316 IF A(UBOUND(O))=0 THEN FOR P=1 TO 99
317 IF A(UBOUND(P))=0 THEN FOR Q=1 TO 99
318 IF A(UBOUND(Q))=0 THEN FOR R=1 TO 99
319 IF A(UBOUND(R))=0 THEN FOR S=1 TO 99
320 IF A(UBOUND(S))=0 THEN FOR T=1 TO 99
321 IF A(UBOUND(T))=0 THEN FOR U=1 TO 99
322 IF A(UBOUND(U))=0 THEN FOR V=1 TO 99
323 IF A(UBOUND(V))=0 THEN FOR W=1 TO 99
324 IF A(UBOUND(W))=0 THEN FOR X=1 TO 99
325 IF A(UBOUND(X))=0 THEN FOR Y=1 TO 99
326 IF A(UBOUND(Y))=0 THEN FOR Z=1 TO 99
327 IF A(UBOUND(Z))=0 THEN FOR A=1 TO 99
328 IF A(UBOUND(A))=0 THEN FOR B=1 TO 99
329 IF A(UBOUND(B))=0 THEN FOR C=1 TO 99
330 IF A(UBOUND(C))=0 THEN FOR D=1 TO 99
331 IF A(UBOUND(D))=0 THEN FOR E=1 TO 99
332 IF A(UBOUND(E))=0 THEN FOR F=1 TO 99
333 IF A(UBOUND(F))=0 THEN FOR G=1 TO 99
334 IF A(UBOUND(G))=0 THEN FOR H=1 TO 99
335 IF A(UBOUND(H))=0 THEN FOR I=1 TO 99
336 IF A(UBOUND(I))=0 THEN FOR J=1 TO 99
337 IF A(UBOUND(J))=0 THEN FOR K=1 TO 99
338 IF A(UBOUND(K))=0 THEN FOR L=1 TO 99
339 IF A(UBOUND(L))=0 THEN FOR M=1 TO 99
340 IF A(UBOUND(M))=0 THEN FOR N=1 TO 99
341 IF A(UBOUND(N))=0 THEN FOR O=1 TO 99
342 IF A(UBOUND(O))=0 THEN FOR P=1 TO 99
343 IF A(UBOUND(P))=0 THEN FOR Q=1 TO 99
344 IF A(UBOUND(Q))=0 THEN FOR R=1 TO 99
345 IF A(UBOUND(R))=0 THEN FOR S=1 TO 99
346 IF A(UBOUND(S))=0 THEN FOR T=1 TO 99
347 IF A(UBOUND(T))=0 THEN FOR U=1 TO 99
348 IF A(UBOUND(U))=0 THEN FOR V=1 TO 99
349 IF A(UBOUND(V))=0 THEN FOR W=1 TO 99
350 IF A(UBOUND(W))=0 THEN FOR X=1 TO 99
351 IF A(UBOUND(X))=0 THEN FOR Y=1 TO 99
352 IF A(UBOUND(Y))=0 THEN FOR Z=1 TO 99
353 IF A(UBOUND(Z))=0 THEN FOR A=1 TO 99
354 IF A(UBOUND(A))=0 THEN FOR B=1 TO 99
355 IF A(UBOUND(B))=0 THEN FOR C=1 TO 99
356 IF A(UBOUND(C))=0 THEN FOR D=1 TO 99
357 IF A(UBOUND(D))=0 THEN FOR E=1 TO 99
358 IF A(UBOUND(E))=0 THEN FOR F=1 TO 99
359 IF A(UBOUND(F))=0 THEN FOR G=1 TO 99
360 IF A(UBOUND(G))=0 THEN FOR H=1 TO 99
361 IF A(UBOUND(H))=0 THEN FOR I=1 TO 99
362 IF A(UBOUND(I))=0 THEN FOR J=1 TO 99
363 IF A(UBOUND(J))=0 THEN FOR K=1 TO 99
364 IF A(UBOUND(K))=0 THEN FOR L=1 TO 99
365 IF A(UBOUND(L))=0 THEN FOR M=1 TO 99
366 IF A(UBOUND(M))=0 THEN FOR N=1 TO 99
367 IF A(UBOUND(N))=0 THEN FOR O=1 TO 99
368 IF A(UBOUND(O))=0 THEN FOR P=1 TO 99
369 IF A(UBOUND(P))=0 THEN FOR Q=1 TO 99
370 IF A(UBOUND(Q))=0 THEN FOR R=1 TO 99
371 IF A(UBOUND(R))=0 THEN FOR S=1 TO 99
372 IF A(UBOUND(S))=0 THEN FOR T=1 TO 99
373 IF A(UBOUND(T))=0 THEN FOR U=1 TO 99
374 IF A(UBOUND(U))=0 THEN FOR V=1 TO 99
375 IF A(UBOUND(V))=0 THEN FOR W=1 TO 99
376 IF A(UBOUND(W))=0 THEN FOR X=1 TO 99
377 IF A(UBOUND(X))=0 THEN FOR Y=1 TO 99
378 IF A(UBOUND(Y))=0 THEN FOR Z=1 TO 99
379 IF A(UBOUND(Z))=0 THEN FOR A=1 TO 99
380 IF A(UBOUND(A))=0 THEN FOR B=1 TO 99
381 IF A(UBOUND(B))=0 THEN FOR C=1 TO 99
382 IF A(UBOUND(C))=0 THEN FOR D=1 TO 99
383 IF A(UBOUND(D))=0 THEN FOR E=1 TO 99
384 IF A(UBOUND(E))=0 THEN FOR F=1 TO 99
385 IF A(UBOUND(F))=0 THEN FOR G=1 TO 99
386 IF A(UBOUND(G))=0 THEN FOR H=1 TO 99
387 IF A(UBOUND(H))=0 THEN FOR I=1 TO 99
388 IF A(UBOUND(I))=0 THEN FOR J=1 TO 99
389 IF A(UBOUND(J))=0 THEN FOR K=1 TO 99
390 IF A(UBOUND(K))=0 THEN FOR L=1 TO 99
391 IF A(UBOUND(L))=0 THEN FOR M=1 TO 99
392 IF A(UBOUND(M))=0 THEN FOR N=1 TO 99
393 IF A(UBOUND(N))=0 THEN FOR O=1 TO 99
394 IF A(UBOUND(O))=0 THEN FOR P=1 TO 99
395 IF A(UBOUND(P))=0 THEN FOR Q=1 TO 99
396 IF A(UBOUND(Q))=0 THEN FOR R=1 TO 99
397 IF A(UBOUND(R))=0 THEN FOR S=1 TO 99
398 IF A(UBOUND(S))=0 THEN FOR T=1 TO 99
399 IF A(UBOUND(T))=0 THEN FOR U=1 TO 99
400 IF A(UBOUND(U))=0 THEN FOR V=1 TO 99
401 IF A(UBOUND(V))=0 THEN FOR W=1 TO 99
402 IF A(UBOUND(W))=0 THEN FOR X=1 TO 99
403 IF A(UBOUND(X))=0 THEN FOR Y=1 TO 99
404 IF A(UBOUND(Y))=0 THEN FOR Z=1 TO 99
405 IF A(UBOUND(Z))=0 THEN FOR A=1 TO 99
406 IF A(UBOUND(A))=0 THEN FOR B=1 TO 99
407 IF A(UBOUND(B))=0 THEN FOR C=1 TO 99
408 IF A(UBOUND(C))=0 THEN FOR D=1 TO 99
409 IF A(UBOUND(D))=0 THEN FOR E=1 TO 99
410 IF A(UBOUND(E))=0 THEN FOR F=1 TO 99
411 IF A(UBOUND(F))=0 THEN FOR G=1 TO 99
412 IF A(UBOUND(G))=0 THEN FOR H=1 TO 99
413 IF A(UBOUND(H))=0 THEN FOR I=1 TO 99
414 IF A(UBOUND(I))=0 THEN FOR J=1 TO 99
415 IF A(UBOUND(J))=0 THEN FOR K=1 TO 99
416 IF A(UBOUND(K))=0 THEN FOR L=1 TO 99
417 IF A(UBOUND(L))=0 THEN FOR M=1 TO 99
418 IF A(UBOUND(M))=0 THEN FOR N=1 TO 99
419 IF A(UBOUND(N))=0 THEN FOR O=1 TO 99
420 IF A(UBOUND(O))=0 THEN FOR P=1 TO 99
421 IF A(UBOUND(P))=0 THEN FOR Q=1 TO 99
422 IF A(UBOUND(Q))=0 THEN FOR R=1 TO 99
423 IF A(UBOUND(R))=0 THEN FOR S=1 TO 99
424 IF A(UBOUND(S))=0 THEN FOR T=1 TO 99
425 IF A(UBOUND(T))=0 THEN FOR U=1 TO 99
426 IF A(UBOUND(U))=0 THEN FOR V=1 TO 99
427 IF A(UBOUND(V))=0 THEN FOR W=1 TO 99
428 IF A(UBOUND(W))=0 THEN FOR X=1 TO 99
429 IF A(UBOUND(X))=0 THEN FOR Y=1 TO 99
430 IF A(UBOUND(Y))=0 THEN FOR Z=1 TO 99
431 IF A(UBOUND(Z))=0 THEN FOR A=1 TO 99
432 IF A(UBOUND(A))=0 THEN FOR B=1 TO 99
433 IF A(UBOUND(B))=0 THEN FOR C=1 TO 99
434 IF A(UBOUND(C))=0 THEN FOR D=1 TO 99
435 IF A(UBOUND(D))=0 THEN FOR E=1 TO 99
436 IF A(UBOUND(E))=0 THEN FOR F=1 TO 99
437 IF A(UBOUND(F))=0 THEN FOR G=1 TO 99
438 IF A(UBOUND(G))=0 THEN FOR H=1 TO 99
439 IF A(UBOUND(H))=0 THEN FOR I=1 TO 99
440 IF A(UBOUND(I))=0 THEN FOR J=1 TO 99
441 IF A(UBOUND(J))=0 THEN FOR K=1 TO 99
442 IF A(UBOUND(K))=0 THEN FOR L=1 TO 99
443 IF A(UBOUND(L))=0 THEN FOR M=1 TO 99
444 IF A(UBOUND(M))=0 THEN FOR N=1 TO 99
445 IF A(UBOUND(N))=0 THEN FOR O=1 TO 99
446 IF A(UBOUND(O))=0 THEN FOR P=1 TO 99
447 IF A(UBOUND(P))=0 THEN FOR Q=1 TO 99
448 IF A(UBOUND(Q))=0 THEN FOR R=1 TO 99
449 IF A(UBOUND(R))=0 THEN FOR S=1 TO 99
450 IF A(UBOUND(S))=0 THEN FOR T=1 TO 99
451 IF A(UBOUND(T))=0 THEN FOR U=1 TO 99
452 IF A(UBOUND(U))=0 THEN FOR V=1 TO 99
453 IF A(UBOUND(V))=0 THEN FOR W=1 TO 99
454 IF A(UBOUND(W))=0 THEN FOR X=1 TO 99
455 IF A(UBOUND(X))=0 THEN FOR Y=1 TO 99
456 IF A(UBOUND(Y))=0 THEN FOR Z=1 TO 99
457 IF A(UBOUND(Z))=0 THEN FOR A=1 TO 99
458 IF A(UBOUND(A))=0 THEN FOR B=1 TO 99
459 IF A(UBOUND(B))=0 THEN FOR C=1 TO 99
460 IF A(UBOUND(C))=0 THEN FOR D=1 TO 99
461 IF A(UBOUND(D))=0 THEN FOR E=1 TO 99
462 IF A(UBOUND(E))=0 THEN FOR F=1 TO 99
463 IF A(UBOUND(F))=0 THEN FOR G=1 TO 99
464 IF A(UBOUND(G))=0 THEN FOR H=1 TO 99
465 IF A(UBOUND(H))=0 THEN FOR I=1 TO 99
466 IF A(UBOUND(I))=0 THEN FOR J=1 TO 99
467 IF A(UBOUND(J))=0 THEN FOR K=1 TO 99
468 IF A(UBOUND(K))=0 THEN FOR L=1 TO 99
469 IF A(UBOUND(L))=0 THEN FOR M=1 TO 99
470 IF A(UBOUND(M))=0 THEN FOR N=1 TO 99
471 IF A(UBOUND(N))=0 THEN FOR O=1 TO 99
472 IF A(UBOUND(O))=0 THEN FOR P=1 TO 99
473 IF A(UBOUND(P))=0 THEN FOR Q=1 TO 99
474 IF A(UBOUND(Q))=0 THEN FOR R=1 TO 99
475 IF A(UBOUND(R))=0 THEN FOR S=1 TO 99
476 IF A(UBOUND(S))=0 THEN FOR T=1 TO 99
477 IF A(UBOUND(T))=0 THEN FOR U=1 TO 99
478 IF A(UBOUND(U))=0 THEN FOR V=1 TO 99
479 IF A(UBOUND(V))=0 THEN FOR W=1 TO 99
480 IF A(UBOUND(W))=0 THEN FOR X=1 TO 99
481 IF A(UBOUND(X))=0 THEN FOR Y=1 TO 99
482 IF A(UBOUND(Y))=0 THEN FOR Z=1 TO 99
483 IF A(UBOUND(Z))=0 THEN FOR A=1 TO 99
484 IF A(UBOUND(A))=0 THEN FOR B=1 TO 99
485 IF A(UBOUND(B))=0 THEN FOR C=1 TO 99
486 IF A(UBOUND(C))=0 THEN FOR D=1 TO 99
487 IF A(UBOUND(D))=0 THEN FOR E=1 TO 99
488 IF A(UBOUND(E))=0 THEN FOR F=1 TO 99
489 IF A(UBOUND(F))=0 THEN FOR G=1 TO 99
490 IF A(UBOUND(G))=0 THEN FOR H=1 TO 99
491 IF A(UBOUND(H))=0 THEN FOR I=1 TO 99
492 IF A(UBOUND(I))=0 THEN FOR J=1 TO 99
493 IF A(UBOUND(J))=0 THEN FOR K=1 TO 99
494 IF A(UBOUND(K))=0 THEN FOR L=1 TO 99
495 IF A(UBOUND(L))=0 THEN FOR M=1 TO 99
496 IF A(UBOUND(M))=0 THEN FOR N=1 TO 99
497 IF A(UBOUND(N))=0 THEN FOR O=1 TO 99
498 IF A(UBOUND(O))=0 THEN FOR P=1 TO 99
499 IF A(UBOUND(P))=0 THEN FOR Q=1 TO 99
500 IF A(UBOUND(Q))=0 THEN FOR R=1 TO 99
501 IF A(UBOUND(R))=0 THEN FOR S=1 TO 99
502 IF A(UBOUND(S))=0 THEN FOR T=1 TO 99
503 IF A(UBOUND(T))=0 THEN FOR U=1 TO 99
504 IF A(UBOUND(U))=0 THEN FOR V=1 TO 99
505 IF A(UBOUND(V))=0 THEN FOR W=1 TO 99
506 IF A(UBOUND(W))=0 THEN FOR X=1 TO 99
507 IF A(UBOUND(X))=0 THEN FOR Y=1 TO 99
508 IF A(UBOUND(Y))=0 THEN FOR Z=1 TO 99
509 IF A(UBOUND(Z))=0 THEN FOR A=1 TO 99
510 IF A(UBOUND(A))=0 THEN FOR B=1 TO 99
511 IF A(UBOUND(B))=0 THEN FOR C=1 TO 99
512 IF A(UBOUND(C))=0 THEN FOR D=1 TO 99
513 IF A(UBOUND(D))=0 THEN FOR E=1 TO 99
514 IF A(UBOUND(E))=0 THEN FOR F=1 TO 99
515 IF A(UBOUND(F))=0 THEN FOR G=1 TO 99
516 IF A(UBOUND(G))=0 THEN FOR H=1 TO 99
517 IF A(UBOUND(H))=0 THEN FOR I=1 TO 99
518 IF A(UBOUND(I))=0 THEN FOR J=1 TO 99
519 IF A(UBOUND(J))=0 THEN FOR K=1 TO 99
520 IF A(UBOUND(K))=0 THEN FOR L=1 TO 99
521 IF A(UBOUND(L))=0 THEN FOR M=1 TO 99
522 IF A(UBOUND(M))=0 THEN FOR N=1 TO 99
523 IF A(UBOUND(N))=0 THEN FOR O=1 TO 99
524 IF A(UBOUND(O))=0 THEN FOR P=1 TO 99
525 IF A(UBOUND(P))=0 THEN FOR Q=1 TO 99
526 IF A(UBOUND(Q))=0 THEN FOR R=1 TO 99
527 IF A(UBOUND(R))=0 THEN FOR S=1 TO 99
528 IF A(UBOUND(S))=0 THEN FOR T=1 TO 99
529 IF A(UBOUND(T))=0 THEN FOR U=1 TO 99
530 IF A(UBOUND(U))=0 THEN FOR V=1 TO 99
531 IF A(UBOUND(V))=0 THEN FOR W=1 TO 99
532 IF A(UBOUND(W))=0 THEN FOR X=1 TO 99
533 IF A(UBOUND(X))=0 THEN FOR Y=1 TO 99
534 IF A(UBOUND(Y))=0 THEN FOR Z=1 TO 99
535 IF A(UBOUND(Z))=0 THEN FOR A=1 TO 99
536 IF A(UBOUND(A))=0 THEN FOR B=1 TO 99
537 IF A(UBOUND(B))=0 THEN FOR C=1 TO 99
538 IF A(UBOUND(C))=0 THEN FOR D=1 TO 99
539 IF A(UBOUND(D))=0 THEN FOR E=1 TO 99
540 IF A(UBOUND(E))=0 THEN FOR F=1 TO 99
541 IF A(UBOUND(F))=0 THEN FOR G=1 TO 99
542 IF A(UBOUND(G))=0 THEN FOR H=1 TO 99
543 IF A(UBOUND(H))=0 THEN FOR I=1 TO 99
544 IF A(UBOUND(I))=0 THEN FOR J=1 TO 99
545 IF A(UBOUND(J))=0 THEN FOR K=1 TO 99
546 IF A(UBOUND(K))=0 THEN FOR L=1 TO 99
547 IF A(UBOUND(L))=0 THEN FOR M=1 TO 99
548 IF A(UBOUND(M))=0 THEN FOR N=1 TO 99
549 IF A(UBOUND(N))=0 THEN FOR O=1 TO 99
550 IF A(UBOUND(O))=0 THEN FOR P=1 TO 99
551 IF A(UBOUND(P))=0 THEN FOR Q=1 TO 99
552 IF A(UBOUND(Q))=0 THEN FOR R=1 TO 99
553 IF A(UBOUND(R))=0 THEN FOR S=1 TO 99
554 IF A(UBOUND(S))=0 THEN FOR T=1 TO 99
555 IF A(UBOUND(T))=0 THEN FOR U=1 TO 99
556 IF A(UBOUND(U))=0 THEN FOR V=1 TO 99
557 IF A(UBOUND(V))=0 THEN FOR W=1 TO 99
558 IF A(UBOUND(W))=0 THEN FOR X=1 TO 99
559 IF A(UBOUND(X))=0 THEN FOR Y=1 TO 99
560 IF A(UBOUND(Y))=0 THEN FOR Z=1 TO 99
561 IF A(UBOUND(Z))=0 THEN FOR A=1 TO 99
562 IF A(UBOUND(A))=0 THEN FOR B=1 TO 99
563 IF A(UBOUND(B))=0 THEN FOR C=1 TO 99
564 IF A(UBOUND(C))=0 THEN FOR D=1 TO 99
565 IF A(UBOUND(D))=0 THEN FOR E=1 TO 99
566 IF A(UBOUND(E))=0 THEN FOR F=1 TO 99
567 IF A(UBOUND(F))=0 THEN FOR G=1 TO 99
568 IF A(UBOUND(G))=0 THEN FOR H=1 TO 99
569 IF A(UBOUND(H))=0 THEN FOR I=1 TO 99
570 IF A(UBOUND(I))=0 THEN FOR J=1 TO 99
571 IF A(UBOUND(J))=0 THEN FOR K=1 TO 99
572 IF A(UBOUND(K))=0 THEN FOR L=1
```

S.P. ELECTRONICS

CPS-80 Dot Matrix Printer, Inc.

Printer Cable £228.00

Parallel Printer Cable £12.90

Canon 160 CPS NLQ Printer,

Inc. Printer Cable £299.00

Green Screen Monitors £85.00

Disk Drives from £120.00

A large range of software available.

SAE for free list.

Large range of spares available.

**Also complete Dragon
repair service**

S.P. ELECTRONICS

68 Limby Road, Hestonall, Notts

(Nottingham NG337T)



0100



HAVE YOU EVER SEEN INSIDE A RAINBOW?



The RAINBOW
(062) 328-6490

The Falcon Building
P.O. Box 385
Prospect, NY 48069

YES! Sign me up for one year (12 issues) of The RAINBOW

Name

Address

City State Zip

Payment enclosed ☐

Charge my: ☐ VISA ☐ MasterCard ☐ American Express

Account #

Cardholder # Exp. Date

Signature

For more information, call (062) 328-6490. For more information, call (062) 328-6490.

You can with a subscription to the RAINBOW, the most complete magazine available for the Color Computer. While the magazine is written for the Tandy CoCo, many of our programs can be typed in with few or no changes for use on the Dragon Computer.

Every month, we bring you an average of 250 pages packed with programs, tutorials, questions and answers, product reviews, hints and tips for expanding your computing horizons. It can all be yours by subscribing to the RAINBOW today!

U.S. Subscription Rates
Surface — \$38 U.S.
Airmail — \$55 U.S.



Subscriptions to THE RAINBOW are \$38 in the United States. Canadian rates \$58 U.S. Single issues sold elsewhere are \$58 U.S. Annual \$115 U.S. All subscriptions begin with the current issue. Please allow 4 to 6 weeks for first copy. Payment accepted in U.S. currency only.



I'VE HAD a letter this month from those two gentlemen from Verona, Marco and Roberto, offering help on El Diablero and others. Where to do the infamous teig? "The oil of cactus blossoms is under the golden statue in the cave, but you'll probably have something else to do before..." And anyone who wants help on Monsters and Magic, Assassin Adventures, Mission of Doom, Danger Island, Don't Panic, Crystal Chamber of Gloom, Franklin's Tomb or El Diablero is invited to write to Marco Pavesio, Via San Rocco 9, 37126 Verona, Italy, or Roberto Vallini, Via Pavetto 5, 37100 Verona, Italy. Maybe they can also give you some tips on where to spend your summer holiday this year.

Frequent updates

Someone who sounds like he needs a holiday in Mr D-King of Basingstoke, who signs himself "Anagard letter", He's tagged because his children have just started playing adventures and he's expected to know all the answers. Now that he's got copies of my clue sheets he's probably a bit better informed, and I think it's about time I informed everyone again as to what's available, as I frequently get letters from people asking for regular updates on this, and I've also just prepared a solution sheet to Hotel Factor, thanks to Simon Hargrave, who's becoming such a regular in this column that it's virtually changing his sex.

Anywyz, in addition to Hotel Factor, for the price it's stamped addressed envelope can also have copies of any or all of the following help sheets/solutions: Black Sanctuary, Cocklewood Incident, El Diablero, Flity Business, Franklin's Tomb, Juxtaposition, Madness and the Minotaur and Theobee: In addition, I've a copy of the notes that were on the original cassette in-fo for El Diablero, and for those who don't want to lose straight into full solutions there's a sheet with all of the various verbs recognised by El Diablero and Theobee.

Chris Morris of Bath couldn't wait with all of those think, but back of S&L means I'll deal with his questions here, as per usual. In Blackwood get past the fireproofed you need TILUMA BHT and you find this on top of HYNCEBIC BHT, while to get over the low you need HOUORHHT OG and then ELLEBUR ISSOPC. For your problem on Verter Factor that won't budge, the answer is that it won't budge. To light the lamp in October Station, the only problem is that the matches

are on one side of the road and the lamp on the other, so S&L and WOHHT and MDM MND, then it's a straightforward PMAL, THUL, and PMAL, THULING. Outside the castle you need to EPCOR WOHHT. For Aquanaut 471 you'll need to write to some clever clogs who's solved it, such as Keith Marshall of 5 Madstone Road, Lowermill, Buffalo NR32 2AY or Stephen McIlhenny of 140 Pembury Road, Penryn, Cornwall CF2 8AP, South Devonport.

Stephen's just written to me to say that he's solved Aquanaut 471, and as well as offering help he would therefore like to swap for an adventure he doesn't have, such as Hotel Factor, Canecon, Monster other graphics adventures apart from Tekboer, Juxtaposition, Syggy, Ring of Darkness or Total Eclipse. He also needs help on destroying Darth Vader. Any offers?

Also in swap mood is Ann Cooper, Ede House, Ingoldmells Holiday Park, Sea Lane, Ingoldmells, Lincoln PE20 1PG. I don't think I'll normally encourage people to go rampantly swapping software instead of buying it, but as Dragon adventures are getting harder to come by all the time, particularly the older titles, I think a bit of trading won't go amiss, as long as it's legitimate swapping of originals and not copies. With adventures in other media is a case of not wanting to play it again once you've cracked it. If you sent S&L to Ann she'll let what she's got, which she says includes several quite old titles plus a few lesser-known American ones. She also offers help on Sea Quest and Shenogans, which gives you some idea of how she might be ready to part with.

Robert Maygreen is ready to part with copies of this adventure, Fernando Pao, having decided to market himself at a price of £3, or £2 if you provide a blank tape and S&L. At either price it's a bargain, and even as repeated I've said in earlier columns which is that Robert's adventure is one of the best I've ever seen sent in by a reader, and it eventually has been published in days when the Dragon was a healthier beast. Pao's address is 24 Canon Young Road, Weymouth, Dorset DT9 1JL.

Phil Dallaghan of Stafford sends in some hints on Tekboer. Can't cross bridge? Trust SCAPS EYOBOR. Can't find Xenator? HOTTUB DER BHT HTW ROOS BHT HOUORHHT. Can't get out of chain after getting out of grass? EERT OF EPCOR BHT.

No clue from Carl Stuart, just the infor-

mation that he's finished and is prepared to help other readers on The Cocklewood Incident, Verter Factor and Aquanaut 471. "Mort is a brilliant adventure," Carl reckons, "I would recommend it to anyone, but I must agree with you about Aquanaut 471 and the irritating 'unclad' screens, nice graphics, but little else." Carl's address for help is 11 Hayden Road, Rushden, Northants NN10 0HX.

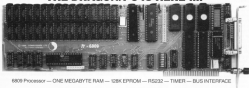
Simon Cocklecock signs himself "A reader without shame," and the reason for this is that he gives various ways of cheating in Richard Shephard's game of Bananagram. Now knowing how many of you readers out there are also without shame, and believing that all is fair in love and sample hunting, here is what Simon says. If you're stuck on a particular level then SPEAK into the program and type LV=7 (ENTER) CONT (ENTER). Instead of the 7 type the number of the level you want to resume playing on. The same method works if you want to change the number of bats that you've killed, in which case type BA=7 and CONTINUE, while you can increase your number of bullets by altering the BL variable. To get a longer look at the map press (SHIFT) + G at the same time to pause and then any key to continue. And what do you need to kill Count Kreeper? The answer's the name of a daily paper, and it isn't the Bananagram Times.

Lost in space

I'd like to thank Bob Wilkinson for sending me four photographic maps of Juxtaposition for passing on to other readers, and there are now four readers somewhere who are grateful to Bob for his kindness. Others will be grateful if they're stuck on that game, Tekboer, Franklin's Tomb, Mansion Adventure or Black Sanctuary next session, as Bob's offering help on those, as well as adventures he's partially completed, like Lost in Space, Flity Business, Pettigrew's Diary, Cocklewood Incident and Syggy. As for one reader's comment that the Adventure Trail heading should be removed to make the column bigger, Bob says "Hands off the Adventure Trail heading is the first thing I look for when I open Dragon User." This man of taste lives at 18 Pound Street, Warrimoor, Wiltshire BA12 8NS.

Another kind creature is H J Newen-Burg, 2 Long Drive, Ambertill, Boston, Lincoln PE20 3PG. This reader is sent in the

THE DRAGON/PC IS HERE !!!!



6809 Processor — ONE MEGABYTE RAM — 128K EPROM — RS232C — TIMER — BUS INTERFACE

Just plug the P16809 into an IBM PC or compatible and you can run FLEX (OS-9 support also available). Use the PC's keyboard, screen printer, disk and share the hard disk. Protect your investment in FLEX and OS-9 software while using the most popular microprocessor business system. CALL OR WRITE FOR DETAILS.

DRAGON/PC Convert

£24.95

Now you can convert your Dragon BASIC programs and data to run on the IBM or compatible (eg Amstrad) Personal Computer PC, XT or AT.

Dragon Basic is very similar to the standard BASICs used on PCs. The hard part is transferring your programs and data to the PC's disk. DRAGON/PC Convert is a program for PCs which reads DRAGON-DOS format disks and copies data and programs to standard MS-DOS format.

Buy an AMSTRAD PC — get DRAGON/PC Convert FREE
From £399.00 (£458.85 inc VAT) plus £300 carriage

DRAGONPLUS

The best upgrade for your DRAGON: test-80 by 34 screen, extra 64K memory, fits inside your DRAGON. Hundreds of satisfied users. Installation service available. Requires 64K DRAGON upgrade service for DRAGON 28K — CALL US. Especially recommended for OS-9 users — get the 80" 34 screen AND save the 5% overhead of the 51 column screen. Use Stylo and C without having to reboot!

DRAGONPLUS

— £100.00 EDIT + CartridgeDisk for DRAGONPLUS

FLEX DRIVERS

— £5.75

OS-9 Drivers

£14.95

£5.75

FLEX SOFTWARE

Get the best out of your DRAGON with FLEX operating system. 51 by 34 screen a standard. 48K free memory for your programs. Includes the best 6809 assembler available for the DRAGON. DBASIC lets you run DRAGON BASIC programs — includes the test boot program. Requires 64K and 05 drive.

FLEX Edit/assembler/DBASIC

— £99.99

DBASIC

— £48.50

SPE word processor spell check

— £85.00

RMS Database

£48.50

***** SPECIAL OFFER — complete package — £199.00 *****

DRAGON HARDWARE — SPECIAL OFFERS

EPROM PROGRAMMER Mark 1 (KIT) for DRAGON 32 reads/programs up to 270K £25.00
EPROM PROGRAMMER Mark 2 (Assembled) reads/programs up to 2725K £35.00
PROTOTYPING BOARD (pin connector) £3.50
PROTOTYPING BOARD CASE £1.50
FLOPPY DISKS (Certified double density) 10 — £9.95 20 — £19.00 50 — £43.00

FREE Cassettes WITH DRAGON DISK DRIVE SYSTEMS

FAST MAIL ORDER — PRICES INCLUDE VAT — PLEASE ADD £2.50 POST & PACKING



FAST MAIL ORDER

Telephone:
01-882 0881 (24 HR)
01-886 4938



COMPUSENSE

PO BOX 189
280C GREEN LANES
PALMER GREEN
LONDON N12 5XA

security code for the Phoenix Software game, The Emperor Must Die, which had what I always thought was a rather silly idea of making you complete an arcade game before you get the pass-code to the adventure in the hint-game pack. If the arcade game's too easy it's a waste of time, and if it's too hard then adventure fans will never get through it. Anyway in that position, please note that the code is printed forward (this time) TCH4. The same reader offers help on *Mister Factor* and would like to hear from anyone who's ever come across any adventures in Dutch.

Andrew Webb of Wells admits that most adventures are double-Dutch to him, and he recognises if there actually is a solution coded for the most talked-about adventures, he'd turn by a mile. He's presently stuck in six of them (that's nothing, Andrew, believe me!), and I've tried to do-sick him in my reply, though he can offer help on the three adventures he's actually solved, *Ring of Darkness*, *Mansion Adventure* and *Darkto Island*. Andrew's partly made progress on *Spygy* by cheating — does this column have no honest readers at all? I've discovered that if you pause the cassette two or three times while loading in a saved game, you can sometimes find yourself in a much better position than you were previously, though this doesn't always work. Andrew's address is 4 20 Catcott Way, Wells, Somerset BA5 3JA.

Karl Emswagan wants to see his name mentioned in *Dragon User*, so obviously the rest of his life is now going to be something of an anti-climax after his moment of glory.

It seems to have an unhealthy interest in obscure adventures (not to mention obscure football teams like Stockport County) and wants to hear from anyone who can help him on *Bliss of Fire* and *To Boldly Go*, and also wonders where he can get hold of copies of two titles from AGI: *Conquering Everest* and *The Valley*. In addition, he can offer help on *Supernap*, *Movie Producer*, *Franklin's Ranch*, *Island Adventure*, *Dragon Mountain*, *Passion Adventure*, *Spygy*, *Castle Adventure* and a typed-in listing called *Adventure* from some little-known rag called *Your Computer*. The Emswagan address is at 31 Blaverton Avenue, Heaton Chapel, Stockport, Cheshire SK4 6JT.

From rather further afield than even Dr.

Stokington comes Guido Jongheis, the Loversvrijstaat 58, 2102 Deurne (Antwerp), Belgium. Guido's also got some pretty obscure adventures, and his latest contribution provides complete details on *Station of Doom*, *Jerusalem Adventure*, *Rescue on Alpha II*, *The Heart of the Desert*, *Ring of Darkness*, *Rescue*, *Spygy* and *The Curse of Abu Simbel*. He's also sent in local tips, and I'll be sending those in future issues — time for a few more clues soon, I feel.

Finally from Richard Heath in Leek, various questions, including what use is the girl in the *Juxtaposition* discotheque? The answer to that is: REINTRAP GIMCMAO DOOG A GEMAM ENG. And that's all, in more ways than one!

Adventure Contact

To help puzzled adventurers further, we are instituting an Adventure Helpline — simply fill in the coupon below, stating the name of the adventure, your problem and your name and address, and send it to Dragon User Adventure Help-

line, 12/13 Little Newport Street, London WC2H 7PP. As space as enough entries have arrived, we will start printing them in the magazine.

Don't worry — you'll still have Adventure Trail to write to as well!

Adventure
Problem

Name
Address

Adventure Contact

Adventure: *Game of Chace*
Problem: How to get past the Temple Key Factory screen
Name: Mabon John
Address: Y Bryn, Bull Lane, Denbigh, Chryd, N. Wales LL28 2SR

Adventure: *Janus's Ultimate*
Problem: Can't do anything, need maps
Name: Todor Gavrilov
Address: 57 Sunnyvale Road, Watton Super Mare, Acon 8520 3AD

Adventure: *Juxtaposition*
Problem: Every time I try to go into Baron White's fort, I get the reply "Not now." Why?
Name: Martin Lewis
Address: 30 London Road, Chichester, Glos. GL1 1RE

Adventure: *Is Mountains of Ice* (3 Temple of Ith)
Problem: Is Weiss are the fast and the clock? Is How can I open the huge door?
Name: Denis Palmier
Address: Via Molina Di Peslaccia 12, 4016 Bologna, Italy

Adventure: *The Vortex Factor*
Problem: How can I open the safe? How can I open the door?
Name: Joseph Jans
Address: Cl Jean Monagall No 5, 2626, 08000 — Villanova, la Galle (Barcelona)

Adventure: *Juxtaposition*
Problem: What use is the girl in the Lasenthaque. How do I get the space from the Droid in the Red Mines
Name: Tim Emswagan
Address: 8 Grappa Close, Wroughton, Swindon, Wiltshire SN4

DISC DRIVE CONTROLLER FOR DRAGONS

fully compatible with DragonDos
 only £69.95 inc. of VAT & post
 Disc drives also available

We also repair Dragons — please enquire.
 Please enclose S.A.E. with any enquiry.
 For further details of DOS see Greener Software Ad.

PNP Communications
 62, Laves Avenue, NEWHAVEN,
 East Sussex, BN9 9SB
 Telephone (0273) 514466

HERE'S MY CLASSIFIED AD.
 (Please write your reply in capitals on the lines below)

Name:
 Address:
 Title:

Please cut out and send this form to: Classified Department, Dragon 1944, 12 — 13 Little Newport St, London WC2H 7PP

TURBOCHARGE YOUR DRAGON:

With our great value software!

BASIC 42 £14.95
An extended BASIC for the Dragon 64 and
DragonDOS (please state version)

Run your Dragon in 64K mode, while retaining BASIC and DOS. Print on the hi-res screen, using all the standard PRINT commands, and a screen layout of 24 rows of 42 columns. Other features include:

Full upper/lower case/graphics character set, REDEFINABLE characters, alternative character sets, CONTROL key for special characters, REPEATING keys and commands in lower case.

8 user-defined, independent WINDOWS, plus 1 fixed, non-destructive window. CATCH command for automatic return to window. INVERTED video (green on black/black on green). True UNDERLINING and extra PRINT commands and functions.

LIBRARY lists commands and functions. Automatic startup of BASIC program. TEXT command for software compatibility. Takes HG-memory from BASIC. Patches for DragonDOS 1.2. Can load in extra UTILITIES from disc.

HELP UTILITY £5.00

Extensions to BASIC 42 include: change CURSOR character, SCROLL, disable, PAUSE listing. Enable/disable repeating keys, enable/disable BREAK key, prevention of disabling of BREAK key, improved TROM (including single stepping). Detailed HELP messages, expanded ERROR messages.

SPOOL UTILITY £5.00

Use computer while printing! 3.5K print buffer. TYPSET program turns Dragon into typewriter.

WHAT THEY SAY ABOUT BASIC 42:
"probably the best step so far Dragon User "as invaluable utility" Dragon Update

SOFTWARE FOR DRAGON 22/64/128 WITH DRAGONDOS/CUMAN DOS 2.0

Professionally written programs for home, clubs, and small business, with random access disk filing and our £2 by 24 screen with true lower case.

MONEYBOX	Personal accounts	£14.95
MAILBOX	Mailing list	£18.95
SHAREBOX	Stocks and shares	£18.95
SALESBOX	Sales Ledger	£19.95
BILLSBOX	Purchase Ledger	£19.95
CASHBOX	Monetary Ledger	£19.95
STOCKSBOX	Stock control	£19.95
ORDERBOX	Invoicing	

Cheque/POs/Further details/Dealer enquiries to:

HARRIS MICRO SOFTWARE
49 Alexandra Road, Hounslow, Middlesex
TW3 4HR. Tel: (01) 570 8335

08/25

BLOCK- BUSTER QUIZ II

FOR THE
DRAGON 64/128

£6.50

MADE
IN
ENGLAND
RECORD

When you purchase this Quiz and
Blockbuster contained in this one popular book
—

Send your name and a cheque or postal
order to the address below. We dispatch
immediately on the same Post Cheque! P.O.
should be made payable to COMPTON STORES.

COMPUTERWARE

P.O. Box 216, Stock-on-Went, ST9 6AR

CLASSIFIED ADS

STOP PLAYING GAMES

Dragonmagazine.com. Dragon
magazine and information centre. Fun and
entertainment available to you in the
computer world. High savings rate.
Includes Dragonmagazine.com and
other links to websites.

Westlink Promotions.
50 George Road, Singapore 050161

Dragon 22 Save down £88 per disk
£150 each. 88 games and software
cartridge £1.50 each or others send
ask for list also books. DJ
magazine etc all offers considered
must sell on Cooper 08575-75421.

Amersham 288 Based machine for
sale £35000 contact Ian Chipper
on 08455 95524.

Shinwa (China) Printer repair. Add
a 4K ram buffer. Easy to follow
instructions — no soldering re-
quired. £7000 inclusive. Stephen
Taylor, 95 Ganger Ave., Chesh-
ington, Surrey, S19 1NR.

Dragon 64, speech-sound, car-
tridge-data recorder, two cassettes,
all tested of Dragon User, seven
books, 1500 instructions, each a
minimum, basic, 6 personal com-
puter, sports mag, 4 other
utilities, 30 games, £1200 open
system 0288-812155.

Dragon 16, disk drive, software,
games £1500 — 020-722 8808.

JOYSTICKS

For Dragon 64/128/256 Joystick. Available
in 6 models and faster. Real stress
control, force as much, £399 each.
Order today on: 02088 501715
to Penwin-Pepi, 62-64 Westfield
Road, London SE10 9BB.

Design or chance

Gordon Lee issues a creative challenge to budding wallpaper artists

THIS month by way of a change the competition is departing from its usual format in favour of something a bit more creative and artistic. In the forthcoming months we will also be including something within the scope of any beginners who have previously fought shy of entering. So, come on all you Dragon users, now's your chance to try for some of those prizes — and don't worry, all you computer masochists, there will still be lots of tougher for you!

Very simply, the competition this month is to design an interesting visual display using one of the high resolution screens of the Dragon. To give you an idea of the sort of thing we're after, try the listing given on this page. As you can see, this program is quite compact, and yet, when run, it is surprising that such an intricate and yet logical design could be produced from this simple process.

This economy of programming will be one of the points that we will be looking for in the entries, which should utilise one of the high resolution screens (PMODE 0 to 4).

What we do not require is a lengthy program which, for example, has virtually every pixel or line drawn and listed within the program itself. You should aim for maximum effect from minimum program!

Your entry must be 'predictable', in the sense that when run it will produce an identical result each time, is not a series of randomly placed circles or rectangles, it is quite in order to use the random feature to control minor details such as colour variations, but the overall design should be 'fixed'.

To enter the competition, send your finished program on a cassette only (no discs please), together with a listing (if possible) and any other documentation relating to your entry. There should be only one program on each cassette, placed at the beginning of the tape — though you may include a duplicated backup copy on the other side. Ensure that your name is marked on the cassette label and, if you would like your tape returned, please enclose a stamped, self addressed envelope.

Finally here is an analysis of the sample program given with this competition. It is based (very loosely) on Pascal's Triangle, a mathematical curiosity which will be dealt with on a future competition page.

As regards the pattern, imagine a rectangular grid 154 squares across by 192 squares down. In each of the squares along the top and left hand edges write in the number 1. Now, starting at the left hand end of the second row, write in each empty square to the immediate left. Continue along the row in this way to the right hand end and then repeat the procedure with

each row in turn down to the bottom.

Of course, if we were to actually do this the numbers on most of the squares would soon become very large, however, if we merely coloured green each square with an odd number in it, and coloured black each square with an even number, the result would be the same as that shown on the screen. The simplicity of the program is dependent on the use of the PPOINT com-

mand to test the colour of the pixels above and to the left of each location, and using this information to decide whether to draw a green or black pixel. The actual numbers are not themselves evaluated, merely whether they are odd (black) or even (green).

That's it! It is a stretch, so now it's up to all of you budding William Morrises to come up with something of your own!

```
10 PMODE4:PCR:SCREEN1:0
20 LINE10,0:-(255,0):PSET
30 LINE10,0:-(0,191):PSET
40 FOR Y=1 TO 191
50 FOR X=1 TO 255
60 P1=PPOINT(X-1,Y):
P2=PPOINT(X,Y-1)
70 IF (P1+P2)/2=INT
(P1+P2)/2)
THEN PSET(X,Y,0)
ELSE PSET(X,Y,1)
80 NEXT X:NEXT Y
90 GOTO 90
```

Prize

"WOTCH!" said Microvision's main man, "We're marketing a few new programs in the new year." "Oh yeah?" we yawned, acting blasé. "How many?" "Seventeen!" he said. So straight away we let off our cheer and touched him for a box of prize yummies. Seventeen is The Random Number, so our random choice was (RANDOM) CHR\$(1). We have twenty Cries to give away to our January prizemen.

Rules

Right. We're having a tiny departure from tradition. Because we realise that not everyone can easily supply a screen dump of their entry, we're allowing commentators to send in cassettes (not discs) as an alternative. Please put in an SAE if you want Gordon to send you cassette back.

There's as usual. Please remember to enclose a listing of your program whenever possible, your name and address, mark your envelope JANUARY COMPETITION, don't forget to post it and — YES! You thought we'd forgotten, didn't you? A TREASURER. Complete the phrase "If seventeen is the Random Number, then..."

October winners

Three winners at Baby Computer Games's BOULDER CRASH! in the October competition are: G. R. Barber of Sutton Coldfield, Christopher James of Tonkin, S. A. Siddiqui of Chislewick, D. Fries Heston (not too many at Denmark), G. C. Heston of Elm, R. J. Taylor of Middleborough, G. Priestland of York, M. Owens of Poynton, Mike Johnson of Newcastle, Keith Davist of Crawley, Denis O'Malley of Comberton, A. Thomas of Stapleford, J. Hewitt of Hixon, D. Denman of Tecklenham, Terry Potter of Chisleham, Michael Graham of Bangor, Mark Draven of Lincoln, Neil Michelson of Norway, Paul Weldon of Wotton Under Edge, and Mark Hooper of Ardenfold.

One or two pretty good tiebreakers, but the favourite comes from S. A. Siddiqui.

"Look out for falling rocks, but don't worry... BF will find you after a few million years."

Solution

OK, OK, you asked for it, and if it makes you at all crawlingly inferior, guilt-ridden, miserable, dolted and puzzled, serve us at night.

See page 5.

NEW RELEASES HALF PRICE!

YES NOW YOU CAN PURCHASE ALL OUR
NEW RELEASES AT LESS THAN HALF
THEIR REGULAR PRICE BY SIMPLY
JOINING OUR **DAY 1 CLUB**

IF YOU HOLD A MAJOR CREDIT CARD THEN YOU SIMPLY FILL
IN THE ORDER FORM BELOW AND WHENEVER WE RELEASE A
NEW LINE FOR THE DRAGON/TANDY COLOUR WE WILL SEND
YOU A COPY AT HALF PRICE ON **DAY 1** AND CHARGE IT TO
YOUR CREDIT CARD! YOU MUST UNDERTAKE TO TAKE AT
LEAST 6 NEW RELEASES BUT AFTER THAT YOU CAN CANCEL
AT ANY TIME.

YOU KNOW AS A PREVIOUS CUSTOMER OF MICRODEALS
THAT ALL OUR PRODUCTS ARE OF HIGH QUALITY AND
WORTH PURCHASING. ALL OUR FUTURE RELEASES WILL
COST £5.95 AND YOU PAY ONLY £2.95 PLUS OUR REGULAR
75p POST & PACKING, SO SIGN UP **NOW!**

I would like to become a member of the **DAY 1** Club and agree to allow Microdeal to charge
£2.95 at regular intervals against my credit card.

Credit Card Type: *Access/Barclaycard/American Express/Diners

Credit Card Number Expiry Date.....
Tick here if you want the latest releases sent straight away ☐

Signed
Date:

Post to: Name



Address

Box 68, St Austell PL25 4YB

Post Code

JOHN PENN
DISCOUNT STORE FOR THE DRAGON

✦ A HAPPY CHRISTMAS TO ALL OUR CUSTOMERS ✦

Game 3

THE THIRTEENTH YEAR: AGI (Age College, Academic, Age, Career, Social, and IT) is the first of three years in the AGI program. The AGI program is a three-year program that is designed to help students develop the skills and knowledge they need to succeed in the 21st century. The AGI program is a three-year program that is designed to help students develop the skills and knowledge they need to succeed in the 21st century.

1000

PHIL BLUMBERG
CONCRETE ATTACK
CONCRETE REPAIRS
NATURAL
CONCRETE

[illegible]

UTILITIES/BUSINESS SOFTWARE

04:00	DOS Software (for use with DOS Operating System, disc drive) and Dragon 44	
04:00	DOS/CALE	
04:00	FILE	025
04:00	BASIC III	025
04:00	T. COMPILER	025
04:00	PASCAL	025
04:00	STOCK RECORDING SYSTEM	025
04:00	Dragon Model A 18T	025

Abstract

[illegible][illegible]

Black & Cherry	1000	125.000	125.000
Black & Cherry (Black)	1000	125.000	125.000

MARGINAL BASEMENT

© 2004 Blackwell Publishing Ltd *Journal of Internal Medicine* 255: 105–112

Please give at least two alternatives when prioritizing the process.

```

(1) (100, 100, 100)
(2) (100, 100, 100)
(3) (100, 100, 100)
(4) (100, 100, 100)
(5) (100, 100, 100)
(6) (100, 100, 100)
(7) (100, 100, 100)
(8) (100, 100, 100)
(9) (100, 100, 100)
(10) (100, 100, 100)
(11) (100, 100, 100)
(12) (100, 100, 100)
(13) (100, 100, 100)
(14) (100, 100, 100)
(15) (100, 100, 100)
(16) (100, 100, 100)
(17) (100, 100, 100)
(18) (100, 100, 100)
(19) (100, 100, 100)
(20) (100, 100, 100)
(21) (100, 100, 100)
(22) (100, 100, 100)
(23) (100, 100, 100)
(24) (100, 100, 100)
(25) (100, 100, 100)
(26) (100, 100, 100)
(27) (100, 100, 100)
(28) (100, 100, 100)
(29) (100, 100, 100)
(30) (100, 100, 100)
(31) (100, 100, 100)
(32) (100, 100, 100)
(33) (100, 100, 100)
(34) (100, 100, 100)
(35) (100, 100, 100)
(36) (100, 100, 100)
(37) (100, 100, 100)
(38) (100, 100, 100)
(39) (100, 100, 100)
(40) (100, 100, 100)
(41) (100, 100, 100)
(42) (100, 100, 100)
(43) (100, 100, 100)
(44) (100, 100, 100)
(45) (100, 100, 100)
(46) (100, 100, 100)
(47) (100, 100, 100)
(48) (100, 100, 100)
(49) (100, 100, 100)
(50) (100, 100, 100)
(51) (100, 100, 100)
(52) (100, 100, 100)
(53) (100, 100, 100)
(54) (100, 100, 100)
(55) (100, 100, 100)
(56) (100, 100, 100)
(57) (100, 100, 100)
(58) (100, 100, 100)
(59) (100, 100, 100)
(60) (100, 100, 100)
(61) (100, 100, 100)
(62) (100, 100, 100)
(63) (100, 100, 100)
(64) (100, 100, 100)
(65) (100, 100, 100)
(66) (100, 100, 100)
(67) (100, 100, 100)
(68) (100, 100, 100)
(69) (100, 100, 100)
(70) (100, 100, 100)
(71) (100, 100, 100)
(72) (100, 100, 100)
(73) (100, 100, 100)
(74) (100, 100, 100)
(75) (100, 100, 100)
(76) (100, 100, 100)
(77) (100, 100, 100)
(78) (100, 100, 100)
(79) (100, 100, 100)
(80) (100, 100, 100)
(81) (100, 100, 100)
(82) (100, 100, 100)
(83) (100, 100, 100)
(84) (100, 100, 100)
(85) (100, 100, 100)
(86) (100, 100, 100)
(87) (100, 100, 100)
(88) (100, 100, 100)
(89) (100, 100, 100)
(90) (100, 100, 100)
(91) (100, 100, 100)
(92) (100, 100, 100)
(93) (100, 100, 100)
(94) (100, 100, 100)
(95) (100, 100, 100)
(96) (100, 100, 100)
(97) (100, 100, 100)
(98) (100, 100, 100)
(99) (100, 100, 100)
(100) (100, 100, 100)

```

[illegible]

EDUCATIONAL CODE: 000000

Copyright © 2004 by
Wiley & Sons
Hoboken, NJ 07030
0471-7553-4

Abstract

Högskolan Högskoleledningen
 Högskoleledningen
 Högskoleledningen

arXiv:1205.0455v1 [math.CO] 2 May 2012

1. 2000年10月1日起，凡在我国境内销售货物的单位和个人，均应按销售额的一定比例缴纳增值税。2000年10月1日起，凡在我国境内销售货物的单位和个人，均应按销售额的一定比例缴纳增值税。

© 2000 Blackwell Science Ltd *Journal of Internal Medicine* 247: 399–406

Algebra Math:
 Quiz Pack
 Smart Pack
 Smart Quizzes
 Integrated : (Graphing Calc series)
 Maths Level 1 (9-10)
 Maths Level 11 (11-12)
 10+ Level Maths :
 Beginner
 Basic Tutor : Beginner's Level
 Basic Tutor : Advanced Level

[illegible]

Keywords: child sexual abuse; disclosure; disclosure strategies

Write to us at photographysuccess@mac.com quoting the title, publisher, price, and location where relevant, of the programs you'd like. Please include \$100 for postage/packaging on single orders, and \$10 for two or more titles ordered. Postage to Europe is \$100 and to the United States is \$10. Please double-check amounts. Please allow four weeks for the book to be sent to you by mail.

If possible please give your telephone number, as well as your name and full address.

Computational order made possible by: xpr / OptiQuest Software. All the software and materials are subject to availability.

Please allow 4-6 days for delivery. Though we try to dispatch orders within 24 hours.

100

THE



DISCOUNT SOFTWARE

Dean Farm Cottage, Kingsley, Bordon, Hants GU24 9AE
Tel Bordon 01493 5479